CSS- Cascading Style Sheets

CSS stands for Cascading Style Sheets.

CSS saves a lot of work. It can control the layout of multiple web pages all at once.

What is CSS?

Cascading Style Sheets (CSS) is used to format the layout of a webpage.

With CSS, you can control the color, font, the size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colors are to be used, different displays for different devices and screen sizes, and much more!

## Using CSS

CSS can be added to HTML documents in 3 ways:

* **Inline** - by using the style attribute inside HTML elements
* **Internal** - by using a <style> element in the <head> section
* **External** - by using a <link> element to link to an external CSS file

The most common way to add CSS, is to keep the styles in external CSS files. However, in this tutorial we will use inline and internal styles, because this is easier to demonstrate, and easier for you to try it yourself.

## Inline CSS

An inline CSS is used to apply a unique style to a single HTML element.

An inline CSS uses the style attribute of an HTML element.

The following example sets the text color of the <h1> element to blue, and the text color of the <p> element to red:

### Example

<h1 style="color:blue;">A Blue Heading</h1>  
  
<p style="color:red;">A red paragraph.</p>

## Internal CSS

An internal CSS is used to define a style for a single HTML page.

An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

The following example sets the text color of ALL the <h1> elements (on that page) to blue, and the text color of ALL the <p> elements to red. In addition, the page will be displayed with a "powderblue" background color:

### Example

<!DOCTYPE html>  
<html>  
<head>  
<style>  
body {background-color: powderblue;}  
h1   {color: blue;}  
p    {color: red;}  
</style>  
</head>  
<body>  
  
<h1>This is a heading</h1>  
<p>This is a paragraph.</p>  
  
</body>  
</html>

## External CSS

An external style sheet is used to define the style for many HTML pages.

To use an external style sheet, add a link to it in the <head> section of each HTML page:

### Example

<!DOCTYPE html>  
<html>  
<head>  
  <link rel="stylesheet" href="styles.css">  
</head>  
<body>  
  
<h1>This is a heading</h1>  
<p>This is a paragraph.</p>  
  
</body>  
</html>

The external style sheet can be written in any text editor. The file must not contain any HTML code, and must be saved with a .css extension.

Here is what the "styles.css" file looks like:

### "styles.css":

body {  
  background-color: powderblue;  
}  
h1 {  
  color: blue;  
}  
p {  
  color: red;  
}

## Link to External CSS

External style sheets can be referenced with a full URL or with a path relative to the current web page.

### Example

This example uses a full URL to link to a style sheet:

<link rel="stylesheet" href="yourfile.css">

# CSS Syntax

A CSS rule consists of a selector and a declaration block.

## CSS Syntax



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

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

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

Multiple CSS declarations are separated with semicolons, and declaration blocks are surrounded by curly braces.

## CSS Colors, Fonts and Sizes

Here, we will demonstrate some commonly used CSS properties. You will learn more about them later.

The CSS color property defines the text color to be used.

The CSS font-family property defines the font to be used.

The CSS font-size property defines the text size to be used.

### Example

Use of CSS color, font-family and font-size properties:

<!DOCTYPE html>  
<html>  
<head>  
<style>  
h1 {  
  color: blue;  
  font-family: verdana;  
  font-size: 300%;  
}  
p {  
  color: red;  
  font-family: courier;  
  font-size: 160%;  
}  
</style>  
</head>  
<body>  
  
<h1>This is a heading</h1>  
<p>This is a paragraph.</p>  
  
</body>  
</html>

## CSS Border

The CSS border property defines a border around an HTML element.

**Tip:** You can define a border for nearly all HTML elements.

### Example

Use of CSS border property:

p {  
  border: 2px solid powderblue;  
}

## CSS Padding

The CSS padding property defines a padding (space) between the text and the border.

### Example

Use of CSS border and padding properties:

p {  
  border: 2px solid powderblue;  
  padding: 30px;  
}

## CSS Margin

The CSS margin property defines a margin (space) outside the border.

### Example

Use of CSS border and margin properties:

p {  
  border: 2px solid powderblue;  
  margin: 50px;  
}

# CSS Selectors

## A CSS selector selects the HTML element(s) you want to style. CSS Selectors

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

We can divide CSS selectors into five categories:

* Simple selectors (select elements based on name, id, class)
* [Combinator selectors](https://www.w3schools.com/css/css_combinators.asp) (select elements based on a specific relationship between them)
* [Pseudo-class selectors](https://www.w3schools.com/css/css_pseudo_classes.asp) (select elements based on a certain state)
* [Pseudo-elements selectors](https://www.w3schools.com/css/css_pseudo_elements.asp) (select and style a part of an element)
* [Attribute selectors](https://www.w3schools.com/css/css_attribute_selectors.asp) (select elements based on an attribute or attribute value).

## The CSS element Selector

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

### Example

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

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

## The CSS id Selector

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

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

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

### Example

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

<!DOCTYPE html>

<html>

<head>

<style>

#para1 {

text-align: center;

color: red;

}

</style>

</head>

<body>

<p id="para1">Hello World!</p>

<p>This paragraph is not affected by the style.</p>

</body>

</html>

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

## The CSS class Selector

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

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

### Example

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

<!DOCTYPE html>

<html>

<head>

<style>

.center {

text-align: center;

color: red;

}

</style>

</head>

<body>

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

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

</body>

</html>

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

### Example

In this example only <p> elements with class="center" will be red and center-aligned:

<!DOCTYPE html>

<html>

<head>

<style>

p.center {

text-align: center;

color: red;

}

</style>

</head>

<body>

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

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

</body>

</html>

## The CSS Universal Selector

The universal selector (\*) selects all HTML elements on the page.

### Example

The CSS rule below will affect every HTML element on the page:

<!DOCTYPE html>

<html>

<head>

<style>

\* {

text-align: center;

color: blue;

}

</style>

</head>

<body>

<h1>Hello world!</h1>

<p>Every element on the page will be affected by the style.</p>

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

<p>And me!</p>

</body>

</html>

## The CSS Grouping Selector

The grouping selector selects all the HTML elements with the same style definitions.

Look at the following CSS code (the h1, h2, and p elements have the same style definitions):

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

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

To group selectors, separate each selector with a comma.

### Example

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

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

## CSS Comments

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

Comments are ignored by browsers.

A CSS comment is placed inside the <style> element, and starts with /\* and ends with \*/

CSS comments are not displayed in the browser, but they can help document your source code.

# CSS Colors

Colors are specified using predefined color names, or RGB, HEX, HSL, RGBA, HSLA values.

## CSS Color Names

In CSS, a color can be specified by using a predefined color name:

Refer this for 140 colors that we use in [csshttps://www.w3schools.com/colors/colors\_names.asp](mailto:csshttps://www.w3schools.com/colors/colors_names.asp?subject=csshttps://www.w3schools.com/colors/colors_names.asp)

## CSS Background Color

You can set the background color for HTML elements:

### Example

<h1 style="background-color:DodgerBlue;">Hello World</h1>  
<p style="background-color:Tomato;">Lorem ipsum...</p>

## CSS Text Color

You can set the color of text:

### Hello World

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

EXAMPLE:

<h1 style="color:Tomato;">Hello World</h1>  
<p style="color:DodgerBlue;">Lorem ipsum...</p>  
<p style="color:MediumSeaGreen;">Ut wisi enim...</p

## CSS Border Color

You can set the color of borders:

## Hello World

## Hello World

## Hello World

### Example

<h1 style="border:2px solid Tomato;">Hello World</h1>  
<h1 style="border:2px solid DodgerBlue;">Hello World</h1>  
<h1 style="border:2px solid Violet;">Hello World</h1>

## CSS background-color

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

### Example

The background color of a page is set like this:

body {  
  background-color: lightblue;  
}

With CSS, a color is most often specified by:

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

## Other Elements

You can set the background color for any HTML elements:

**<!DOCTYPE html>**

**<html>**

**<head>**

**<style>**

**h1 {**

**background-color: green;**

**}**

**div {**

**background-color: lightblue;**

**}**

**p {**

**background-color: yellow;**

**}**

**</style>**

**</head>**

**<body>**

**<h1>CSS background-color example!</h1>**

**<div>**

**This is a text inside a div element.**

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

**We are still in the div element.**

**</div>**

**</body>**

**</html>**

## Opacity / Transparency

The opacity property specifies the opacity/transparency of an element. It can take a value from 0.0 - 1.0. The lower value, the more transparent:

opacity 1

opacity 0.6

opacity 0.3

opacity 0.1

### Example

**<!DOCTYPE html>**

**<html>**

**<head>**

**<style>**

**div {**

**background-color: green;**

**}**

**div.first {**

**opacity: 0.1;**

**}**

**div.second {**

**opacity: 0.3;**

**}**

**div.third {**

**opacity: 0.6;**

**}**

**</style>**

**</head>**

**<body>**

**<h1>Transparent Boxes</h1>**

**<p>When using the opacity property to add transparency to the background of an element, all of its child elements become transparent as well. This can make the text inside a fully transparent element hard to read:</p>**

**<div class="first">**

**<h1>opacity 0.1</h1>**

**</div>**

**<div class="second">**

**<h1>opacity 0.3</h1>**

**</div>**

**<div class="third">**

**<h1>opacity 0.6</h1>**

**</div>**

**<div>**

**<h1>opacity 1 (default)</h1>**

**</div>**

**</body>**

**</html>**

# CSS Borders

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

I have borders on all sides.

I have a red bottom border.

I have rounded borders.

CSS Border Style

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

The following values are allowed:

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

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

### Example

Demonstration of the different border styles:

p.dotted {border-style: dotted;}  
p.dashed {border-style: dashed;}  
p.solid {border-style: solid;}  
p.double {border-style: double;}  
p.groove {border-style: groove;}  
p.ridge {border-style: ridge;}  
p.inset {border-style: inset;}  
p.outset {border-style: outset;}  
p.none {border-style: none;}  
p.hidden {border-style: hidden;}  
p.mix {border-style: dotted dashed solid double;}

Result:

A dotted border.

A dashed border.

A solid border.

A double border.

A groove border. The effect depends on the border-color value.

A ridge border. The effect depends on the border-color value.

An inset border. The effect depends on the border-color value.

An outset border. The effect depends on the border-color value.

No border.

A hidden border.

A mixed border.

## CSS Border Width

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

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

### xample

Demonstration of the different border widths:

p.one {  
  border-style: solid;  
  border-width: 5px;  
}  
  
p.two {  
  border-style: solid;  
  border-width: medium;  
}  
  
p.three {  
  border-style: dotted;  
  border-width: 2px;  
}  
  
p.four {  
  border-style: dotted;  
  border-width: thick;  
}

## CSS Border Color

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

The color can be set by:

* name - specify a color name, like "red"
* HEX - specify a HEX value, like "#ff0000"
* RGB - specify a RGB value, like "rgb(255,0,0)"
* HSL - specify a HSL value, like "hsl(0, 100%, 50%)"
* transparent

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

### Example

Demonstration of the different border colors:

**<!DOCTYPE html>**

**<html>**

**<head>**

**<style>**

**p.one {**

**border-style: solid;**

**border-color: red;**

**}**

**p.two {**

**border-style: solid;**

**border-color: green;**

**}**

**p.three {**

**border-style: dotted;**

**border-color: blue;**

**}**

**</style>**

**</head>**

**<body>**

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

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

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

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

**<p class="three">A dotted blue border</p>**

**<p class="three">A dotted blue border</p>**

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

**</body>**

**</html>**

**Specific Side Colors**

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

EXAMPLE:

<!DOCTYPE html>

<html>

<head>

<style>

p.one {

border-style: solid;

border-color: red green blue yellow; /\* red top, green right, blue bottom and yellow left \*/

}

</style>

</head>

<body>

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

<p>The border-color property can have from one to four values (for the top border, right border, bottom border, and the left border):</p>

<p class="one">A solid multicolor border</p>

</body>

</html>

## CSS Rounded Borders

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

Normal border

Round border

Rounder border

Roundest border

<!DOCTYPE html>

<html>

<head>

<style>

p.normal {

border: 2px solid red;

padding: 5px;

}

p.round1 {

border: 2px solid red;

border-radius: 5px;

padding: 5px;

}

p.round2 {

border: 2px solid red;

border-radius: 8px;

padding: 5px;

}

p.round3 {

border: 2px solid red;

border-radius: 12px;

padding: 5px;

}

</style>

</head>

<body>

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

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

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

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

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

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

</body>

</html>

## All CSS Border Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| [border](https://www.w3schools.com/cssref/pr_border.asp) | Sets all the border properties in one declaration |
| [border-bottom](https://www.w3schools.com/cssref/pr_border-bottom.asp) | Sets all the bottom border properties in one declaration |
| [border-bottom-color](https://www.w3schools.com/cssref/pr_border-bottom_color.asp) | Sets the color of the bottom border |
| [border-bottom-style](https://www.w3schools.com/cssref/pr_border-bottom_style.asp) | Sets the style of the bottom border |
| [border-bottom-width](https://www.w3schools.com/cssref/pr_border-bottom_width.asp) | Sets the width of the bottom border |
| [border-color](https://www.w3schools.com/cssref/pr_border-color.asp) | Sets the color of the four borders |
| [border-left](https://www.w3schools.com/cssref/pr_border-left.asp) | Sets all the left border properties in one declaration |
| [border-left-color](https://www.w3schools.com/cssref/pr_border-left_color.asp) | Sets the color of the left border |
| [border-left-style](https://www.w3schools.com/cssref/pr_border-left_style.asp) | Sets the style of the left border |
| [border-left-width](https://www.w3schools.com/cssref/pr_border-left_width.asp) | Sets the width of the left border |
| [border-radius](https://www.w3schools.com/cssref/css3_pr_border-radius.asp) | Sets all the four border-\*-radius properties for rounded corners |
| [border-right](https://www.w3schools.com/cssref/pr_border-right.asp) | Sets all the right border properties in one declaration |
| [border-right-color](https://www.w3schools.com/cssref/pr_border-right_color.asp) | Sets the color of the right border |
| [border-right-style](https://www.w3schools.com/cssref/pr_border-right_style.asp) | Sets the style of the right border |
| [border-right-width](https://www.w3schools.com/cssref/pr_border-right_width.asp) | Sets the width of the right border |
| [border-style](https://www.w3schools.com/cssref/pr_border-style.asp) | Sets the style of the four borders |
| [border-top](https://www.w3schools.com/cssref/pr_border-top.asp) | Sets all the top border properties in one declaration |
| [border-top-color](https://www.w3schools.com/cssref/pr_border-top_color.asp) | Sets the color of the top border |
| [border-top-style](https://www.w3schools.com/cssref/pr_border-top_style.asp) | Sets the style of the top border |
| [border-top-width](https://www.w3schools.com/cssref/pr_border-top_width.asp) | Sets the width of the top border |
| [border-width](https://www.w3schools.com/cssref/pr_border-width.asp) | Sets the width of the four borders |

W3schools Pathfinder

Track your progress - it's free!

# CSS Margins

Margins are used to create space around elements, outside of any defined borders.

CSS Margins

The CSS margin properties are used to create space around elements, outside of any defined borders.

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

Margin - Individual Sides

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

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

All the margin properties can have the following values:

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

**Tip:** Negative values are allowed.

Example:

<!DOCTYPE html>

<html>

<head>

<style>

div {

border: 1px solid black;

margin-top: 100px;

margin-bottom: 100px;

margin-right: 150px;

margin-left: 80px;

background-color: lightblue;

}

</style>

</head>

<body>

<h2>Using individual margin properties</h2>

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

</body>

</html>

# CSS Padding

Padding is used to create space around an element's content, inside of any defined borders.

CSS Padding

The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

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

Padding - Individual Sides

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

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

All the padding properties can have the following values:

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

<!DOCTYPE html>

<html>

<head>

<style>

div {

border: 1px solid black;

background-color: lightblue;

padding-top: 50px;

padding-right: 30px;

padding-bottom: 50px;

padding-left: 80px;

}

</style>

</head>

<body>

<h2>Using individual padding properties</h2>

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

</body>

</html>

# CSS Height, Width and Max-width

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

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

CSS Setting height and width

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

The height and width properties do not include padding, borders, or margins. It sets the height/width of the area inside the padding, border, and margin of the element.

CSS height and width Values

The height and width properties may have the following values:

* auto - This is default. The browser calculates the height and width
* length - Defines the height/width in px, cm, etc.
* % - Defines the height/width in percent of the containing block
* initial - Sets the height/width to its default value
* inherit - The height/width will be inherited from its parent value

CSS height and width Examples

**<!DOCTYPE html>**

**<html>**

**<head>**

**<style>**

**div {**

**height: 200px;**

**width: 50%;**

**background-color: powderblue;**

**}**

**</style>**

**</head>**

**<body>**

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

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

**</body>**

**</html>**

# CSS Text

## CSS has a lot of properties for formatting textText Color

The color property is used to set the color of the text. The color is specified by:

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

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

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

### Example

body {  
  color: blue;  
}  
  
h1 {  
  color: green;  
}

## Text Color and Background Color

In this example, we define both the background-color property and the color property:

### Example

body {  
  background-color: lightgrey;  
  color: blue;  
}  
  
h1 {  
  background-color: black;  
  color: white;  
}  
  
div {  
  background-color: blue;  
  color: white;  
}

# CSS Text Alignment

## Text Alignment and Text Direction

In this chapter you will learn about the following properties:

* text-align
* text-align-last
* direction
* unicode-bidi
* vertical-align

## Text Alignment

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

A text can be left or right aligned, centered, or justified.

The following example shows center aligned, and left and right aligned text (left alignment is default if text direction is left-to-right, and right alignment is default if text direction is right-to-left):

### Example

h1 {  
  text-align: center;  
}  
  
h2 {  
  text-align: left;  
}  
  
h3 {  
  text-align: right;  
}

When the text-align property 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

div {  
  text-align: justify;  
}

## Text Align Last

The text-align-last property specifies how to align the last line of a text.

### Example

Align the last line of text in three <p> elements:

p.a {  
  text-align-last: right;  
}  
  
p.b {  
  text-align-last: center;  
}  
  
p.c {  
  text-align-last: justify;  
}

## Text Decoration

In this chapter you will learn about the following properties:

* text-decoration-line
* text-decoration-color
* text-decoration-style
* text-decoration-thickness
* text-decoration

Add a Decoration Line to Text

The text-decoration-line property is used to add a decoration line to text.

**Tip:** You can combine more than one value, like overline and underline to display lines both over and under a text.

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

## Specify a Color for the Decoration Line

The text-decoration-color property is used to set the color of the decoration line.

### Example

h1 {  
  text-decoration-line: overline;  
  text-decoration-color: red;  
}  
  
h2 {  
  text-decoration-line: line-through;  
  text-decoration-color: blue;  
}  
  
h3 {  
  text-decoration-line: underline;  
  text-decoration-color: green;  
}  
  
p {  
  text-decoration-line: overline underline;  
  text-decoration-color: purple;  
}

## Specify a Style for the Decoration Line

The text-decoration-style property is used to set the style of the decoration line.

### Example

h1 {  
  text-decoration-line: underline;  
  text-decoration-style: solid;  
}  
  
h2 {  
  text-decoration-line: underline;  
  text-decoration-style: double;  
}  
  
h3 {  
  text-decoration-line: underline;  
  text-decoration-style: dotted;  
}  
  
p.ex1 {  
  text-decoration-line: underline;  
  text-decoration-style: dashed;  
}  
  
p.ex2 {  
  text-decoration-line: underline;  
  text-decoration-style: wavy;  
}  
  
p.ex3 {  
  text-decoration-line: underline;  
  text-decoration-color: red;  
  text-decoration-style: wavy;  
}

## Specify the Thickness for the Decoration Line

The text-decoration-thickness property is used to set the thickness of the decoration line.

### Example

h1 {  
  text-decoration-line: underline;  
  text-decoration-thickness: auto;  
}  
  
h2 {  
  text-decoration-line: underline;  
  text-decoration-thickness: 5px;  
}  
  
h3 {  
  text-decoration-line: underline;  
  text-decoration-thickness: 25%;  
}  
  
p {  
  text-decoration-line: underline;  
  text-decoration-color: red;  
  text-decoration-style: double;  
  text-decoration-thickness: 5px;  
}

# CSS Text Transformation

## 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

<!DOCTYPE html>

<html>

<head>

<style>

p.uppercase {

text-transform: uppercase;

}

p.lowercase {

text-transform: lowercase;

}

p.capitalize {

text-transform: capitalize;

}

</style>

</head>

<body>

<h1>Using the text-transform property</h1>

<p class="uppercase">This text is transformed to uppercase.</p>

<p class="lowercase">This text is transformed to lowercase.</p>

<p class="capitalize">This text is capitalized.</p>

</body>

</html>

## Text Spacing

In this chapter you will learn about the following properties:

* text-indent
* letter-spacing
* line-height
* word-spacing
* white-space

## Text Indentation

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

### Example

p {  
  text-indent: 50px;  
}

## Letter Spacing

The letter-spacing property is used to specify the space between the characters in a text.

The following example demonstrates how to increase or decrease the space between characters:

### Example

h1 {  
  letter-spacing: 5px;  
}  
  
h2 {  
  letter-spacing: -2px;  
}

## Line Height

The line-height property is used to specify the space between lines:

### Example

<!DOCTYPE html>

<html>

<head>

<style>

p.small {

line-height: 0.7;

}

p.big {

line-height: 1.8;

}

</style>

</head>

<body>

<h1>Using line-height</h1>

<p>

This is a paragraph with a standard line-height.<br>

The default line height in most browsers is about 110% to 120%.<br>

</p>

<p class="small">

This is a paragraph with a smaller line-height.<br>

This is a paragraph with a smaller line-height.<br>

</p>

<p class="big">

This is a paragraph with a bigger line-height.<br>

This is a paragraph with a bigger line-height.<br>

</p>

</body>

</html>

## Word Spacing

The word-spacing property is used to specify the space between the words in a text.

The following example demonstrates how to increase or decrease the space between words:

<!DOCTYPE html>

<html>

<head>

<style>

p.one {

word-spacing: 10px;

}

p.two {

word-spacing: -2px;

}

</style>

</head>

<body>

<h1>Using word-spacing</h1>

<p>This is a paragraph with normal word spacing.</p>

<p class="one">This is a paragraph with larger word spacing.</p>

<p class="two">This is a paragraph with smaller word spacing.</p>

</body>

</html>

## White Space

The white-space property specifies how white-space inside an element is handled.

This example demonstrates how to disable text wrapping inside an element:

### Example

<!DOCTYPE html>

<html>

<head>

<style>

p {

white-space: nowrap;

}

</style>

</head>

<body>

<h1>Using white-space</h1>

<p>

This is some text that will not wrap.

This is some text that will not wrap.

This is some text that will not wrap.

This is some text that will not wrap.

This is some text that will not wrap.

This is some text that will not wrap.

This is some text that will not wrap.

This is some text that will not wrap.

This is some text that will not wrap.

</p>

<p>Try to remove the white-space property to see the difference!</p>

</body>

</html>

# CSS Text Shadow

## Text Shadow

The text-shadow property adds shadow to text.

In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px):

## Text shadow effect!

### Example

h1 {  
  text-shadow: 2px 2px;  
}

Next, add a color (red) to the shadow:

## Text shadow effect!

<!DOCTYPE html>

<html>

<head>

<style>

h1 {

text-shadow: 2px 2px red;

}

</style>

</head>

<body>

<h1>Text-shadow effect!</h1>

</body>

</html>

Then, add a blur effect (5px) to the shadow:

h1 {  
  text-shadow: 2px 2px 5px red;  
}

## More Text Shadow Examples

### Example 1

Text-shadow on a white text:

h1 {  
  color: white;  
  text-shadow: 2px 2px 4px #000000;  
}

# CSS Fonts

Choosing the right font for your website is important!

Font Selection is Important

Choosing the right font has a huge impact on how the readers experience a website.

The right font can create a strong identity for your brand.

Using a font that is easy to read is important. The font adds value to your text. It is also important to choose the correct color and text size for the font.

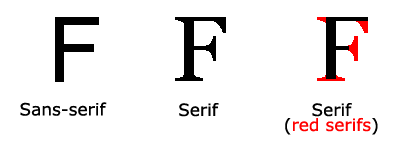
Generic Font Families

In CSS there are five generic font families:

1. **Serif** fonts have a small stroke at the edges of each letter. They create a sense of formality and elegance.
2. **Sans-serif** fonts have clean lines (no small strokes attached). They create a modern and minimalistic look.
3. **Monospace** fonts - here all the letters have the same fixed width. They create a mechanical look.
4. **Cursive** fonts imitate human handwriting.
5. **Fantasy** fonts are decorative/playful fonts.

All the different font names belong to one of the generic font families.

Difference Between Serif and Sans-serif Fonts



**Note:** On computer screens, sans-serif fonts are considered easier to read than serif font

## Some Font Examples

|  |  |
| --- | --- |
| **Generic Font Family** | **Examples of Font Names** |
| Serif | Times New Roman Georgia Garamond |
| Sans-serif | Arial Verdana Helvetica |
| Monospace | Courier New Lucida Console Monaco |
| Cursive | Brush Script MT Lucida Handwriting |
| Fantasy | Copperplate Papyrus |

ADVERTISEMENT

## The CSS font-family Property

In CSS, we use the font-family property to specify the font of a text.

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

**Tip:** The font-family property should hold several font names as a "fallback" system, to ensure maximum compatibility between browsers/operating systems. 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). The font names should be separated with comma.

<!DOCTYPE html>

<html>

<head>

<style>

.p1 {

font-family: "Times New Roman", Times, serif;

}

.p2 {

font-family: Arial, Helvetica, sans-serif;

}

.p3 {

font-family: "Lucida Console", "Courier New", monospace;

}

</style>

</head>

<body>

<h1>CSS font-family</h1>

<p class="p1">This is a paragraph, shown in the Times New Roman font.</p>

<p class="p2">This is a paragraph, shown in the Arial font.</p>

<p class="p3">This is a paragraph, shown in the Lucida Console font.</p>

</body>

</html>

# CSS Web Safe Fonts

## What are Web Safe Fonts?

Web safe fonts are fonts that are universally installed across all browsers and devices.

## Fallback Fonts

However, there are no 100% completely web safe fonts. There is always a chance that a font is not found or is not installed properly.

Therefore, it is very important to always use fallback fonts.

This means that you should add a list of similar "backup fonts" in the font-family property. If the first font does not work, the browser will try the next one, and the next one, and so on. Always end the list with a generic font family name.

### Example

Here, there are three font types: Tahoma, Verdana, and sans-serif. The second and third fonts are backups, in case the first one is not found.

p {  
font-family: Tahoma, Verdana, sans-serif;  
}

Best Web Safe Fonts for HTML and CSS

The following list are the best web safe fonts for HTML and CSS:

* Arial (sans-serif)
* Verdana (sans-serif)
* Tahoma (sans-serif)
* Trebuchet MS (sans-serif)
* Times New Roman (serif)
* Georgia (serif)
* Garamond (serif)
* Courier New (monospace)
* Brush Script MT (cursive)

**Note:** Before you publish your website, always check how your fonts appear on different browsers and devices

# CSS Font Style

## 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;  
}

## Font Variant

The font-variant property specifies whether or not a text should be displayed in a small-caps font.

In a small-caps font, all lowercase letters are converted to uppercase letters. However, the converted uppercase letters appears in a smaller font size than the original uppercase letters in the text.

### Example

<!DOCTYPE html>

<html>

<head>

<style>

p.normal {

font-variant: normal;

}

p.small {

font-variant: small-caps;

}

</style>

</head>

<body>

<h1>The font-variant property</h1>

<p class="normal">My name is Hege Refsnes.</p>

<p class="small">My name is Hege Refsnes.</p>

</body>

</html>

# CSS Font Size

## 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

**Note:** 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;  
}

## Set Font Size With Em

To allow users to resize the text (in the browser menu), many developers use em instead of pixels.

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 \*/  
}

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 Internet Explorer. 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;  
}

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

## Google Fonts

If you do not want to use any of the standard fonts in HTML, you can use Google Fonts.

Google Fonts are free to use, and have more than 1000 fonts to choose from.

## How To Use Google Fonts

Just add a special style sheet link in the <head> section and then refer to the font in the CSS.

### Example

Here, we want to use a font named "Sofia" from Google Fonts:

<head>  
**<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Sofia">**  
<style>

body {  
  font-family: "Sofia", sans-serif;  
}  
</style>  
</head>

# CSS Icons

Icons can easily be added to your HTML page, by using an icon library.

## How To Add Icons

The simplest way to add an icon to your HTML page, is with an icon library, such as Font Awesome.

Add the name of the specified icon class to any inline HTML element (like <i> or <span>).

All the icons in the icon libraries below, are scalable vectors that can be customized with CSS (size, color, shadow, etc.)

## Font Awesome Icons

To use the Font Awesome icons, go to [fontawesome.com](https://fontawesome.com/), sign in, and get a code to add in the <head> section of your HTML page:

<script src="https://kit.fontawesome.com/yourcode.js" crossorigin="anonymous"></script>

**Note:** No downloading or installation is required!

<!DOCTYPE html>

<html>

<head>

<title>Font Awesome Icons</title>

<meta name="viewport" content="width=device-width, initial-scale=1">

<script src="https://kit.fontawesome.com/a076d05399.js" crossorigin="anonymous"></script>

<!--Get your own code at fontawesome.com-->

</head>

<body>

<h1>Font Awesome icon library</h1>

<p>Some Font Awesome icons:</p>

<i class="fas fa-cloud"></i>

<i class="fas fa-heart"></i>

<i class="fas fa-car"></i>

<i class="fas fa-file"></i>

<i class="fas fa-bars"></i>

<p>Styled Font Awesome icons (size and color):</p>

<i class="fas fa-cloud" style="font-size:24px;"></i>

<i class="fas fa-cloud" style="font-size:36px;"></i>

<i class="fas fa-cloud" style="font-size:48px;color:red;"></i>

<i class="fas fa-cloud" style="font-size:60px;color:lightblue;"></i>

</body>

</html>

## Styling Links

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

<!DOCTYPE html>

<html>

<head>

<style>

a {

color: hotpink;

}

</style>

</head>

<body>

<h2>Style a link with a color</h2>

<p><b><a href="default.asp" target="\_blank">This is a link</a></b></p>

</body>

</html>

## Link Buttons

This example demonstrates a more advanced example where we combine several CSS properties to display links as boxes/buttons:

### Example

<!DOCTYPE html>

<html>

<head>

<style>

a:link, a:visited {

background-color: #f44336;

color: white;

padding: 14px 25px;

text-align: center;

text-decoration: none;

display: inline-block;

}

a:hover, a:active {

background-color: red;

}

</style>

</head>

<body>

<h2>Link Button</h2>

<p>A link styled as a button:</p>

<a href="default.asp" target="\_blank">This is a link</a>

</body>

</html>

# CSS Lists

## Unordered Lists:

* Coffee
* Tea
* Coca Cola
* Coffee
* Tea
* Coca Cola

## Ordered Lists:

1. Coffee
2. Tea
3. Coca Cola
4. Coffee
5. Tea
6. Coca Cola

## HTML Lists and CSS List Properties

In HTML, there are two main types of lists:

* unordered lists (<ul>) - the list items are marked with bullets
* ordered lists (<ol>) - the list items are marked with numbers or letters

The CSS list properties allow you to:

* Set different list item markers for ordered lists
* Set different list item markers for unordered lists
* Set an image as the list item marker
* Add background colors to lists and list items

## Different List Item Markers

The list-style-type property specifies the type of list item marker.

The following example shows some of the available list item markers:

### Example

!DOCTYPE html>

<html>

<head>

<style>

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;

}

</style>

</head>

<body>

<h2>The list-style-type Property</h2>

<p>Example of unordered lists:</p>

<ul class="a">

<li>Coffee</li>

<li>Tea</li>

<li>Coca Cola</li>

</ul>

<ul class="b">

<li>Coffee</li>

<li>Tea</li>

<li>Coca Cola</li>

</ul>

<p>Example of ordered lists:</p>

<ol class="c">

<li>Coffee</li>

<li>Tea</li>

<li>Coca Cola</li>

</ol>

<ol class="d">

<li>Coffee</li>

<li>Tea</li>

<li>Coca Cola</li>

</ol>

</body>

</html>

## Styling List With Colors

We can also style lists with colors, to make them look a little more interesting.

Anything added to the <ol> or <ul> tag, affects the entire list, while properties added to the <li> tag will affect the individual list items:

### Example

<!DOCTYPE html>

<html>

<head>

<style>

ol {

background: #ff9999;

padding: 20px;

}

ul {

background: #3399ff;

padding: 20px;

}

ol li {

background: #ffe5e5;

color: darkred;

padding: 5px;

margin-left: 35px;

}

ul li {

background: #cce5ff;

color: darkblue;

margin: 5px;

}

</style>

</head>

<body>

<h1>Styling Lists With Colors</h1>

<ol>

<li>Coffee</li>

<li>Tea</li>

<li>Coca Cola</li>

</ol>

<ul>

<li>Coffee</li>

<li>Tea</li>

<li>Coca Cola</li>

</ul>

</body>

</html>

# CSS Tables

## The look of an HTML table can be greatly improved with CSS: Table Borders

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

The example below specifies a solid border for <table>, <th>, and <td> elements:

|  |  |
| --- | --- |
| **Firstname** | **Lastname** |
| Peter | Griffin |
| Lois | Griffin |

<!DOCTYPE html>

<html>

<head>

<style>

table, th, td {

border: 1px solid;

}

</style>

</head>

<body>

<h2>Add a border to a table:</h2>

<table>

<tr>

<th>Firstname</th>

<th>Lastname</th>

</tr>

<tr>

<td>Peter</td>

<td>Griffin</td>

</tr>

<tr>

<td>Lois</td>

<td>Griffin</td>

</tr>

</table>

</body>

</html>

## Full-Width Table

The table above might seem small in some cases. If you need a table that should span the entire screen (full-width), add width: 100% to the <table> element:

|  |  |
| --- | --- |
| **Firstname** | **Lastname** |
| Peter | Griffin |
| Lois | Griffin |

### Example

table {  
  width: 100%;  
}

### Double Borders

Notice that the table in the examples above have double borders. This is because both the table and the <th> and <td> elements have separate borders.

To remove double borders, take a look at the example below.

## Collapse Table Borders

The border-collapse property sets whether the table borders should be collapsed into a single border:

|  |  |
| --- | --- |
| **Firstname** | **Lastname** |
| Peter | Griffin |
| Lois | Griffin |

### Example

table {  
  border-collapse: collapse;  
}

If you only want a border around the table, only specify the border property for <table>:

|  |  |
| --- | --- |
| **Firstname** | **Lastname** |
| Peter | Griffin |
| Lois | Griffin |

### Example

<!DOCTYPE html>

<html>

<head>

<style>

table {

width: 100%;

border: 1px solid;

}

</style>

</head>

<body>

<h2>Single Border Around The Table</h2>

<table>

<tr>

<th>Firstname</th>

<th>Lastname</th>

</tr>

<tr>

<td>Peter</td>

<td>Griffin</td>

</tr>

<tr>

<td>Lois</td>

<td>Griffin</td>

</tr>

</table>

</body>

</html>

# CSS Table Size

## Table Width and Height

The width and height of a table are 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 70px:

|  |  |  |
| --- | --- | --- |
| **Firstname** | **Lastname** | **Savings** |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

### Example

table {  
  width: 100%;  
}  
  
th {  
  height: 70px;  
}

To create a table that should only span half the page, use width: 50%:

|  |  |  |
| --- | --- | --- |
| **Firstname** | **Lastname** | **Savings** |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

# CSS Table Alignment

## Horizontal Alignment

The text-align property sets the horizontal alignment (like left, right, or center) of the content in <th> or <td>.

By default, the content of <th> elements are center-aligned and the content of <td> elements are left-aligned.

To center-align the content of  <td> elements as well, use text-align: center:

|  |  |  |
| --- | --- | --- |
| **Firstname** | **Lastname** | **Savings** |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

<!DOCTYPE html>

<html>

<head>

<style>

table, td, th {

border: 1px solid black;

}

table {

border-collapse: collapse;

width: 100%;

}

td {

text-align: center;

}

</style>

</head>

<body>

<h2>The text-align Property</h2>

<p>This property sets the horizontal alignment (like left, right, or center) of the content in th or td.</p>

<table>

<tr>

<th>Firstname</th>

<th>Lastname</th>

<th>Savings</th>

</tr>

<tr>

<td>Peter</td>

<td>Griffin</td>

<td>$100</td>

</tr>

<tr>

<td>Lois</td>

<td>Griffin</td>

<td>$150</td>

</tr>

<tr>

<td>Joe</td>

<td>Swanson</td>

<td>$300</td>

</tr>

<tr>

<td>Cleveland</td>

<td>Brown</td>

<td>$250</td>

</tr>

</table>

</body>

</html>

To left-align the content, force the alignment of <th> elements to be left-aligned, with the text-align: left property:

|  |  |  |
| --- | --- | --- |
| **Firstname** | **Lastname** | **Savings** |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

### Example

th {  
  text-align: left;  
}

## Vertical Alignment

The vertical-align property sets the vertical alignment (like top, bottom, or middle) of the content in <th> or <td>.

By default, the vertical alignment of the content in a table is middle (for both <th> and <td> elements).

The following example sets the vertical text alignment to bottom for <td> elements:

|  |  |  |
| --- | --- | --- |
| **Firstname** | **Lastname** | **Savings** |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

### Example

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

## Table Padding

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

| **First Name** | **Last Name** | **Savings** |
| --- | --- | --- |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

### Example

th, td {  
  padding: 15px;  
  text-align: left;  
}

## Horizontal Dividers

| **First Name** | **Last Name** | **Savings** |
| --- | --- | --- |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

Add the border-bottom property to <th> and <td> for horizontal dividers:

### Example

th, td {  
  border-bottom: 1px solid #ddd;  
}

## 

## Striped Tables

|  |  |  |
| --- | --- | --- |
| **First Name** | **Last Name** | **Savings** |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:

### Example

<!DOCTYPE html>

<html>

<head>

<style>

table {

border-collapse: collapse;

width: 100%;

}

th, td {

text-align: left;

padding: 8px;

}

tr:nth-child(even) {

background-color: #f2f2f2;

}

</style>

</head>

<body>

<h2>Striped Table</h2>

<p>For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:</p>

<table>

<tr>

<th>First Name</th>

<th>Last Name</th>

<th>Points</th>

</tr>

<tr>

<td>Peter</td>

<td>Griffin</td>

<td>$100</td>

</tr>

<tr>

<td>Lois</td>

<td>Griffin</td>

<td>$150</td>

</tr>

<tr>

<td>Joe</td>

<td>Swanson</td>

<td>$300</td>

</tr>

<tr>

<td>Cleveland</td>

<td>Brown</td>

<td>$250</td>

</tr>

</table>

</body>

</html>

## Table Color

The example below specifies the background color and text color of <th> elements:

|  |  |  |
| --- | --- | --- |
| **First Name** | **Last Name** | **Savings** |
| Peter | Griffin | $100 |
| Lois | Griffin | $150 |
| Joe | Swanson | $300 |

### Example

th {  
  background-color: #04AA6D;  
  color: white;  
}

# CSS Layout - Overflow

The CSS overflow property controls what happens to content that is too big to fit into an area.

This text is really long and the height of its container is only 100 pixels. Therefore, a scrollbar is added to help the reader to scroll the content. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. Typi non habent claritatem insitam; est usus legentis in iis qui facit eorum claritatem.

[Try it Yourself »](https://www.w3schools.com/css/tryit.asp?filename=trycss_overflow_intro)

## CSS Overflow

The overflow property specifies whether to clip the content or to add scrollbars when the content of an element is too big to fit in the specified area.

The overflow property has the following values:

* visible - Default. The overflow is not clipped. The content renders outside the element's box
* hidden - The overflow is clipped, and the rest of the content will be invisible
* scroll - The overflow is clipped, and a scrollbar is added to see the rest of the content
* auto - Similar to scroll, but it adds scrollbars only when necessary

**Note:** The overflow property only works for block elements with a specified height.

**Note:** In OS X Lion (on Mac), scrollbars are hidden by default and only shown when being used (even though "overflow:scroll" is set).

<!DOCTYPE html>

<html>

<head>

<style>

#overflowTest {

background: #4CAF50;

color: white;

padding: 15px;

width: 50%;

height: 100px;

overflow: scroll;

border: 1px solid #ccc;

}

</style>

</head>

<body>

<h2>CSS Overflow</h2>

<p>The overflow property controls what happens to content that is too big to fit into an area.</p>

<div id="overflowTest">This text is really long and the height of its container is only 100 pixels. Therefore, a scrollbar is added to help the reader to scroll the content. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. Typi non habent claritatem insitam; est usus legentis in iis qui facit eorum claritatem.</div>

</body>

</html>

# CSS Navigation Bar

## 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:

## <!DOCTYPE html>

## <html>

## <body>

## <ul>

## <li><a href="#home">Home</a></li>

## <li><a href="#news">News</a></li>

## <li><ahref="#contact">Contact</a></li>

## <li><a href="#about">About</a></li>

## </ul>

## <p>Note: We use href="#" for test links. In a real web site this would be URLs.</p>

## </body>

## </html>

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

### Example

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

}

</style>

</head>

<body>

<p>In this example, we remove the bullets from the list, and its default padding and margin.</p>

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

</body>

</html>

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

The code in the example above is the standard code used in both vertical, and horizontal navigation bars, which you will learn more about in the next chapters.

# CSS Vertical Navigation Bar

## Vertical Navigation Bar

* [Home](javascript:void(0))
* [News](javascript:void(0))
* [Contact](javascript:void(0))
* [About](javascript:void(0))

To build a vertical navigation bar, you can style the <a> elements inside the list, in addition to the code from the previous page:

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

}

li a {

display: block;

width: 60px;

background-color: #dddddd;

}

</style>

</head>

<body>

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

<p>A background color is added to the links to show the link area.</p>

<p>Notice that the whole link area is clickable, not just the text.</p>

</body>

</html>

You can also set the width of <ul>, and remove the width of <a>, as they will take up the full width available when displayed as block elements. This will produce the same result as our previous example:

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

width: 60px;

}

li a {

display: block;

background-color: #dddddd;

}

</style>

</head>

<body>

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

<p>A background color is added to the links to show the link area.</p>

<p>Notice that the whole link area is clickable, not just the text.</p>

</body>

</html>

## Vertical Navigation Bar Examples

Create a basic vertical navigation bar with a gray background color and change the background color of the links when the user moves the mouse over them:

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

width: 200px;

background-color: #f1f1f1;

}

li a {

display: block;

color: #000;

padding: 8px 16px;

text-decoration: none;

}

/\* Change the link color on hover \*/

li a:hover {

background-color: #555;

color: white;

}

</style>

</head>

<body>

<h2>Vertical Navigation Bar</h2>

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

</body>

</html>

### Active/Current Navigation Link

Add an "active" class to the current link to let the user know which page he/she is on:

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

width: 200px;

background-color: #f1f1f1;

}

li a {

display: block;

color: #000;

padding: 8px 16px;

text-decoration: none;

}

li a.active {

background-color: #04AA6D;

color: white;

}

li a:hover:not(.active) {

background-color: #555;

color: white;

}

</style>

</head>

<body>

<h2>Vertical Navigation Bar</h2>

<p>In this example, we create an "active" class with a green background color and a white text. The class is added to the "Home" link.</p>

<ul>

<li><a class="active" href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

</body>

</html>

### Full-height Fixed Vertical Navbar

<!DOCTYPE html>

<html>

<head>

<style>

body {

margin: 0;

}

ul {

list-style-type: none;

margin: 0;

padding: 0;

width: 25%;

background-color: #f1f1f1;

position: fixed;

height: 100%;

overflow: auto;

}

li a {

display: block;

color: #000;

padding: 8px 16px;

text-decoration: none;

}

li a.active {

background-color: #04AA6D;

color: white;

}

li a:hover:not(.active) {

background-color: #555;

color: white;

}

</style>

</head>

<body>

<ul>

<li><a class="active" href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

<div style="margin-left:25%;padding:1px 16px;height:1000px;">

<h2>Fixed Full-height Side Nav</h2>

<h3>Try to scroll this area, and see how the sidenav sticks to the page</h3>

<p>Notice that this div element has a left margin of 25%. This is because the side navigation is set to 25% width. If you remove the margin, the sidenav will overlay/sit on top of this div.</p>

<p>Also notice that we have set overflow:auto to sidenav. This will add a scrollbar when the sidenav is too long (for example if it has over 50 links inside of it).</p>

<p>Some text..</p>

<p>Some text..</p>

<p>Some text..</p>

<p>Some text..</p>

<p>Some text..</p>

<p>Some text..</p>

<p>Some text..</p>

</div>

</body>

</html>

# CSS Horizontal Navigation Bar

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

### 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 from the previous page:

### Example

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

}

li {

display: inline;

}

</style>

</head>

<body>

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

</body>

</html>

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

### Floating List Items

Another way of creating a horizontal navigation bar is to float the <li> elements, and specify a layout for the navigation links:

### Example

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

overflow: hidden;

}

li {

float: left;

}

li a {

display: block;

padding: 8px;

background-color: #dddddd;

}

</style>

</head>

<body>

<ul>

<li><a href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

<p><b>Note:</b> If a !DOCTYPE is not specified, floating items can produce unexpected results.</p>

<p>A background color is added to the links to show the link area. The whole link area is clickable, not just the text.</p>

<p><b>Note:</b> overflow:hidden is added to the ul element to prevent li elements from going outside of the list.</p>

</body>

</html>

Example explained:

* float: left; - Use float to get block elements to float next to each other
* display: block; - Allows us to specify padding (and height, width, margins, etc. if you want)
* padding: 8px; - Specify some padding between each <a> element, to make them look good
* background-color: #dddddd; - Add a gray background-color to each <a> element

**Tip:** Add the background-color to <ul> instead of each <a> element if you want a full-width background color:

ul {  
  background-color: #dddddd;  
}

## Horizontal Navigation Bar Examples

Create a basic horizontal navigation bar with a dark background color and change the background color of the links when the user moves the mouse over them:

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

overflow: hidden;

background-color: #333;

}

li {

float: left;

}

li a {

display: block;

color: white;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

li a:hover {

background-color: #111;

}

</style>

</head>

<body>

<ul>

<li><a class="active" href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

</body>

</html>

### Active/Current Navigation Link

Add an "active" class to the current link to let the user know which page he/she is on:

### Example

.active {  
  background-color: #04AA6D;  
}

FULL EXAMPLE:

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

overflow: hidden;

background-color: #333;

}

li {

float: left;

}

li a {

display: block;

color: white;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

li a:hover:not(.active) {

background-color: #111;

}

.active {

background-color: #04AA6D;

}

</style>

</head>

<body>

<ul>

<li><a class="active" href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li><a href="#about">About</a></li>

</ul>

</body>

</html>

### Border Dividers

Add the border-right property to <li> to create link dividers:

### Example

/\* Add a gray right border to all list items, except the last item (last-child) \*/  
li {  
  border-right: 1px solid #bbb;  
}  
  
li:last-child {  
  border-right: none;  
}

FULL EXAMPLE:

<!DOCTYPE html>

<html>

<head>

<style>

ul {

list-style-type: none;

margin: 0;

padding: 0;

overflow: hidden;

background-color: #333;

}

li {

float: left;

border-right:1px solid #bbb;

}

li:last-child {

border-right: none;

}

li a {

display: block;

color: white;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

li a:hover:not(.active) {

background-color: #111;

}

.active {

background-color: #04AA6D;

}

</style>

</head>

<body>

<ul>

<li><a class="active" href="#home">Home</a></li>

<li><a href="#news">News</a></li>

<li><a href="#contact">Contact</a></li>

<li style="float:right"><a href="#about">About</a></li>

</ul>

</body>

</html>