

COMMON CSS PROPERTIES

CSS Comment	<code>/* comment */</code>
CSS Rule Syntax	<code>selector { property: value; }</code>
Example: Targets all <p> tags to change text color to red	<code>p { color: red; }</code>
FONT PROPERTIES	
Specify font for an element	<code>font-family: "Times New Roman", Times, serif; font-family:</code>
Set font size for an element	<code>font-size: 12px; font-size: [large small]; font-size: 200%;</code>
Set font to normal or italic	<code>font-style: italic; font-style: normal;</code>
Set font to normal or bold	<code>font-weight: bold; font-weight: normal;</code>
Set to small-caps or normal	<code>font-variant: small-caps; font-variant: normal;</code>
Shorthand	<code>font: style, variant, weight, size/height, family</code>
TEXT PROPERTIES	
Set the color of text	<code>color: red;</code>
Set space between characters	<code>letter-spacing: normal;</code>
Set space between lines	<code>line-height: 1.6px; line-height: 80%;</code>
Specify horizontal alignment of text	<code>text-align: center; text-align: right; text-align: left; text-align: justified;</code>
Specify decoration applied to text elements	<code>text-decoration: none; text-decoration: overline; text-decoration: underline; text-decoration: overline underline; text-decoration: line-through;</code>
Set text to uppercase, lowercase or capitalize	<code>text-transform: uppercase; text-transform: lowercase; text-transform: capitalize;</code>
Set how text overflow displays	<code>text-overflow: clip; text-overflow: hidden;</code>

BACKGROUND PROPERTIES	
Set the background color	background-color: red;
Set image as a background	background-image: url("path/to/img.jpg");
Set starting position of background image	background-position: [center right left]; background-position: [x% y%][x_pos y_pos];
Set size of background	background-size: auto width height cover contain
Set repeat of image	background-repeat: no-repeat repeat repeat-y repeat-x
Background shorthand	background: color, image, position, size, repeat, origin, clip, attachment
BORDER PROPERTIES	
Set width for border lines	border-width: [medium thin thick px]
Set style of border lines	border-style: none hidden dotted dashed double single groove ridge inset outset initial inherit
Set color of border	border-color: red;
Set radius of border	border-radius: 50px;
Border shorthand property	border: width, style, color
SPACING PROPERTIES	
Padding: space between content and border (Same values for margin)	padding: all-sides padding: top-bottom, right-left padding: top, right-left, bottom padding: top, right, bottom, left
Space between elements	margin: top, right, bottom, left
SIZING PROPERTIES	
Set height of an element	height: [auto initial inherit px % of parent]
Set width of an element	width: [auto initial inherit px % of parent]
Set maximum width/height	max-width: [none length initial inherit]; max-height: [none length initial inherit];
Set minimum width/height:	min-height: [none length initial inherit]; min-width: [none length initial inherit];
LIST PROPERTIES	
Set style of list marker	list-style-type: [circle square initial inherit]
Set position of list markers	list-style-position: [outside inside initial]
Set image as list marker	list-style-image: url('/path/to/img.jpg');
Shorthand for list	list-style: type, position, image

CSS COLORS

Specify using color name	<code>color: red;</code> <code>color: darkgrey;</code> <code>color: MediumSeaGreen;</code>
Specify using rgb: Example:	<code>rgb(red, green, blue)</code> <code>color: rgb(255,0,255)</code> <code>color: rgb(50, 100, 150)</code> <code>rgb(90, 90, 90)</code>
Specify using rgba: A = Alpha: 0: fully transparent 1: fully opaque	<code>rgba(red, green, blue, alpha)</code> <code>color: rgba(255, 0, 0, 0.3);</code> <code>color: rgba(255, 0, 0, 1);</code>
Specify using hexadecimal Example:	<code>#rrggbb</code> <code>color: 009900;</code> <code>color: ff22aa;</code>
Three-digit hex code: Equivalent to ff0000: Equivalent to ffcc99:	<code>#rgb</code> <code>color: f00;</code> <code>color: #fc9;</code>
Specify using hsl Hue = degree from 0 – 360 0: red, 120: green, 240: blue Saturation = 0 – 100% 0 = grey, 100 = full color Lightness = 0 – 100% 0 = black, 100% = white	<code>hsl(hue, saturation, lightness)</code> <code>color: hsl(0, 100%, 50%);</code> <code>color: hsl(100, 50%, 25%);</code> <code>color: hsl(240, 75%, 50%);</code> <code>color: hsl(360, 50%, 60%);</code> <code>color: hsl(300, 20%, 50%);</code>

CSS UNITS OF MEASUREMENT

Relative in CSS2, absolute in CSS3 (1/96th of an inch)	px
Value of the font-size property of the element it is used on.	em
Value relative to the parent element	%
Value relative to the x-height of the current font	ex
Value relative to width of "0" (zero) of the current font	ch
Value relative to font-size of the root element	rem
Value from 0 – 100, relative to 1% of the width/height of viewport	vw/vh
Absolute value of an inch, centimeter and millimeter	in/cm/mm
Absolute value of a point (pt = 1/72 of an inch) & pica (pc = 1/6 of an inch)	pt/pc

SELECTORS

Universal (all) selector	<code>* { ... }</code>
Element selector: Selects all p tags	<code>p { ... }</code>
Grouped selector: Selects all p, div, and h1 tags	<code>p, div, h1 { ... }</code>
Class selector: Selects all elements with class name 'cl1'	<code>.cl1 { ... }</code>
Selects only p elements with the class name 'c1'	<code>p.cl1 { ... }</code>
Id selector: Selects element with id name 'id1'	<code>#id1 { ... }</code>
Selects only a div element with the id name 'id1'	<code>div#id1 { ... }</code>
Attribute selector: Selects elements with attribute	<code>[attribute]</code>
Example: Targets elements with href attribute	HTML: <code>Link</code> CSS: <code>[href] { ... }</code>
Select specific tag with given attribute	<code>tag[attribute]</code>
Example: Matches img tags with alt attribute	<code>img[alt]</code>
Matches attributes set to value	<code>[attribute="value"]</code>
<code>~=</code> : Matches attribute value containing 'example'	<code>[attribute~="example"]</code>
<code>^=</code> : Matches attribute value beginning with 'example'	<code>[attribute^="test"]</code>
<code>\$=</code> : Matches attributes value ending with 'example'	<code>[attribute\$="test"]</code>

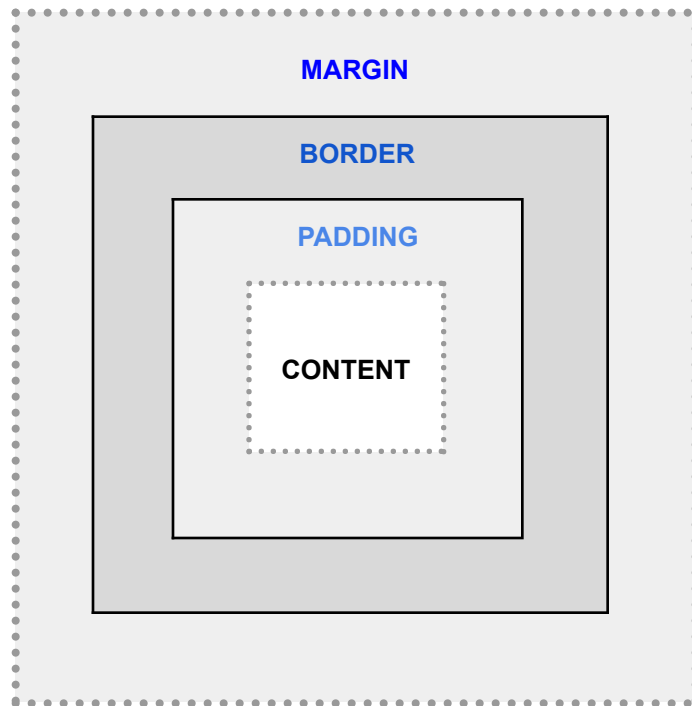
combinators

Descendant: All elements contained within another	<code>parent descendant { ... }</code>
Example: Selects all <p> elements within <div> elements	<code>div p { ... }</code>
Child: Element that is a direct child of specified element	<code>parent>child { ... }</code>
Example: Selects <p> elements that are directly inside div	<code>div>p { ... }</code>
Adjacent Sibling: Targets elements directly after another	<code>sib1+sib2 { ... }</code>
Example: Selects all <h2> tags directly after <h1> tags	<code>h2+h1 { ... }</code>
General Sibling: Targets all elements after another	<code>sib1~sib2 { ... }</code>
Example: Selects all <p> tags after <h2> tags	<code>h2~p { ... }</code>

pseudo selectors

Syntax	selector: pseudo-class { property: value; }
Hover: Targets elements with mouseover	element:hover
Example: Targets all p tags with the class 'target' when they are being hovered over	p.target:hover { ... }
Example: Targets all p tags within a div when the div element is being hovered over	div:hover p { ... }
<a> tag: Unvisited link Visited link Mouse-over link Selected (click-down) link	a:link a:visited a:hover /* after link & visited */ a:active /* after hover */
<input> tag: Input elements that are checked Input elements that are disabled Input elements that are enabled (default) Input elements that have focus (clicked on) Input elements within a specific range Input elements with an invalid value Input elements with a valid value Input elements with a required attribute Input elements without a required attribute	input:checked input:disabled input:enabled input:focus input:in-range input:invalid input:valid input:required input:optional
Elements that are the first child	element:first-child
Example: Matches first <i> inside a <p> tag	p i:first-child
Elements that are the last child	element:last-child
Elements that are the only child of its parent	element:only-child
Every child element whose index is odd	element:nth-child(odd)
Every child element whose index is even	element:nth-child(even)
Every <p> element that is the first <p> element of its parent	p:first-of-type
Every <p> element that is the last <p> element of its parent	p:last-of-type
Every <p> element that is the only <p> element of its parent	p:only-of-type

BOX MODEL



ELEMENT WIDTH:

[LeftMargin](#)
+ [LeftBorder](#)
+ [LeftPadding](#)
+ **Width of Content**
+ [RightPadding](#)
+ [RightBorder](#)
+ [RightMargin](#)

ELEMENT HEIGHT:

[TopMargin](#)
+ [TopBorder](#)
+ [TopPadding](#)
+ **Height of Content**
+ [BottomPadding](#)
+ [BottomBorder](#)
+ [BottomMargin](#)

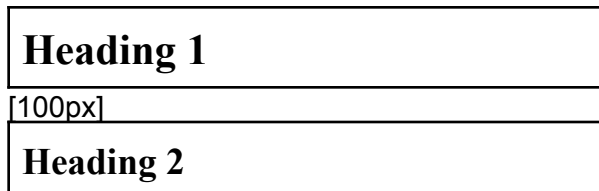
COLLAPSE

margin-collapse

The margin between two elements will be the larger of the two margins, not the sum.

The resulting vertical space between h1 and h2 will be 100px, not 200px:

```
h1 {  
  border: 1px solid black;  
  margin: 100px;  
}  
  
h2 {  
  border: 1px solid black;  
  margin: 100px;  
}
```



border-collapse

The border in a table can be set to be separate or a single, collapsed border:

```
border-collapse: separate; /* separate borders for cells and table */  
border-collapse: collapse; /* single border for cells and table */
```

INHERITANCE

UNIVERSAL PROPERTY VALUES	
Sets value to that of its parent element	<code>inherit</code>
Sets value to the initial value of that property	<code>initial</code>
Resets value to the browser's default style	<code>revert</code>
Resets value to one specified in previous cascade layer	<code>revert-layer</code>
Resets to natural value (inherited or initial)	<code>unset</code>

cascading rules & specificity

BROWSER	Browser default style	<code><!-- basic style defined by browser --></code>
USER STYLE	Overrides browser	<code><!-- user-defined stylesheet by web user --></code>
EXTERNAL	Overrides user style and previously defined stylesheets	<code><link rel="stylesheet" href="style1.css" /></code> <code><link rel="stylesheet" href="style2.css" /></code>
INTERNAL	Overrides External	<code><style></code> <code>p { color: blue; }</code> <code></style></code>
INLINE	Overrides Internal	<code><p style="color:red"></code>
SPECIFICITY		
Element	Most recently defined overrides others	<code>div { color: red; }</code> <code>div { color: blue; }</code>
Class/Pseudo Class	Overrides Element	<code>.container { color: green; }</code>
Id	Overrides Class	<code>#nav { color: yellow; }</code>

scoring specificity

A - B - C

- Add 1 to **A** for each **ID** in selector
- Add 1 to **B** for each **class** or pseudo-class in selector
- Add 1 to **C** for each **element** name
- Read result as a 3-digit number

Example: `<div id="nav" class="links">`
 `div { color: green; }` `/* 0 0 1 */`
 `div#nav { color: red; }` `/* 1 0 1 */` -> div will be red
 `div.links { color: blue; }` `/* 0 1 1 */`

POSITION & DISPLAY

float: Used for positioning an element within its container	
The element floats to the left of its container	<code>float: left;</code>
The element floats to the right of its container	<code>float: right;</code>
The element does not float (default)	<code>float: none;</code>
The element inherits the float value of its parent	<code>float: inherit;</code>
Example: The image will float to the left of its container and other elements will be positioned to the right of it	<pre>img { float: left; }</pre>
clear: Specifies position of the element that is next to a floating element.	
The element is not pushed below left or right (default)	<code>clear: none;</code>
The element is pushed below left floated elements	<code>clear: left;</code>
The element is pushed below right floated elements	<code>clear: right;</code>
The element is pushed below left & right floated elements	<code>clear: both;</code>
The element inherits the clear value from its parent	<code>clear: inherit;</code>
Example: The content of div1 will be floated to the left of its container. The content of div2 will be cleared below div1. Without this, the content of each div could overlap.	<pre>.div1 { float: left; } .div2 { clear: left; }</pre>
position	
top, right, bottom, left properties define offset (Example: element will be 10px offset from expected top position)	<code>top: 10px;</code> <code>right: 30px;</code>
Positioned according to the flow of the page (default)	<code>position: static;</code>
Positioned relative to its normal position	<code>position: relative;</code>
Positioned in a fixed location relative to viewport	<code>position: fixed;</code>
Positioned relative to the nearest positioned ancestor	<code>position: absolute;</code>
Positioned based on the user's scroll position.	<code>position: sticky;</code>
Example: div is positioned in its normal location, when the user scrolls down, it will become fixed at the top location (top: 0 means 0 pixels from the top of the viewport)	<pre>div.sticky { position: sticky; top: 0; }</pre>

z-index: Greater z-index of overlapped elements will appear on top	
Example: Without z-index, last defined element appears on top	<code><div>Below</div> <div>In Front</div></code>
z-index changes the order of which elements appear first	<code>.behind { z-index: -1; } .middle { z-index: 0; } .infront { z-index: 1; }</code>
display	
Hide element (will show blank space in HTML page)	<code>visibility: hidden;</code>
Hide element (no blank space in HTML page)	<code>display: none;</code>
Change element to be displayed as block element	<code>display: block;</code>
Change element to be displayed as inline element	<code>display: inline;</code>