

Selectors		
Style x element(s) with y class	p.first	x.y
Style selector with x tag	a[href]	s[x]
Style selector with x tag with exact y content	a[href="http://google.com"]	s[x="y"]
Style selector with x tag containing y	a[href="google"]	s[x*="y"]
Style selector with x tag starting by y	a[href^="https"]	s[x^="y"]
Style selector with x tag ending by y	a[href\$="org"]	s[x\$="y"]
Style selector with x tag starting by y and ending by z	a[href="https://org"]	s[x^="y" & \$="z"]
Style descendant selectors (element inside element)	.class div	s x
Style direct descendant selectors (direct element child inside element)	.class > div	s > x
Style direct sibling to element	.class + div	s + x
Select direct siblings after element	.class ~ div	s ~ x

Concepts

Selectors (cont)		
Specificity or weight	If multiple rules target the same element, the browser will apply the most specific.	ID selector > class, attribute selectors > element selector
	To overrule specificity, !important can be used	color: royalblue !important;
	Combining id and class will give us higher specificity	.highlight#product

Pseudo-class selectors	
Style first child of x element	x :first-child
Style last child of x element	x :last-child
Style each first x child element of each type	x :first-of-type
Style each last x child element of each type	x :last-of-type
Style odd child elements of x	x y:nth-child(odd)
Style visited URL	a:visited
Style all link anchors	a:link
Style element when hovered over	x:hover
Style element when selected	x:focus

Pseudo-element selectors	
Style first letter inside element	p::first-letter
Style first line inside element	p::first-line
Change selection to another color	::selection

Pseudo-element selectors (cont)	
Change selection of element to another color	p::selection
Insert content before element and incrust into DOM	::before { content: "..."; display: block}
Insert content after element and incrust into DOM	::after { content: "...";}

Inheritance	
Child elements automatically inherit some styles from parents.	
To stop from doing this, we use the 'initial' property on the child	color: initial;
Styles that aren't inherited can be forced to be, such as border with 'inherit'.	border: inherit;

Colors		
RGB	Amount of Red Green and Blue	hsl( 1, 2, 3)
RGBA	Amount of Red Green and Blue with added Alpha (amount of transparency)	hsl( 1, 2, 3, 4%)
HEX	Value representing numbers in the RGB system	#abc123
HSL	Hue (tone) Saturation (percentage of saturation) and Lighting (the amount of lightness)	hsl( 1, 2%, 3%)



By **raposinha**  
[cheatography.com/raposinha/](https://cheatography.com/raposinha/)

Not published yet.  
 Last updated 8th January, 2024.  
 Page 1 of 2.

Sponsored by **Readable.com**  
 Measure your website readability!  
<https://readable.com>

### Colors (cont)

HSLA Hue (tone) Saturation hsl( (percentage of saturation) and Lighting (the amount of lightness), Alpha (Transparency) 4)

### Gradients

They're technically images, so the properties used for colors don't work the same

Fill background: linear-gradient( (blue,red); with a linear gradient

From left to right background: linear-gradient(to right, blue,red);

From bottom right background: linear-gradient(to bottom right, blue,red);

background: linear-gradient(45deg, blue,red);

Same linear-gradient(45deg, blue,red 90%); thing, mostly red

3 color linear-gradient(45deg, blue, red, yellow);

Radial radial-gradient(blue, red) gradient

Radial radial-gradient(circle, blue, red) gradient

with rounder center

Radial radial-gradient(circle at top left, blue, red)

with circle centre located at the top left

Websites to generate gradients [cssgradient.io](https://cssgradient.io)

### Borders

border: 10px line width solid, line style dotted, dashed, blue color

border-top

border-bottom

The order of the border lines matters: top, right, bottom, left border-width: 10px 20px 10px 30px

If we don't set left, left will inherit right. If we don't set top, bottom will inherit top. And viceversa.

Set a style for each line in borders border-style: dotted, dashed

Set a color for each line in borders border-color: red, blue, yellow;

Set roundness of borders. If absolute a circle is created. border-radius px, %

We can make even more shapes, just look up CSS Shapes

### Shadows

The box-shadow property can be used to create shadows around elements. box-shadow

Positive values will shift the shadow to the right. box-shadow: 10px;

Negative values will shift the shadow to the left. box-shadow: -10px;

Two positive values will shift the shadow to the right and down. box-shadow: 10px 10px;

### Shadows (cont)

One positive value and one negative value will shift the shadow to the right and left. box-shadow: 10px -10px;

One negative value and one positive value will shift the shadow to the left and right. box-shadow: -10px 10px;

To change the color of the shadow, just add it after size box-shadow: -10px 10px grey;

To change the degradation or blur of the color of the shadow, add a value before color color box-shadow: 10px 10px 10px grey;

For a soft central shadow under a shape, use this trick: box-shadow: 0 0 30px grey;

We can also use shadows for text

We use the property text-shadow just like box-shadow text-shadow: 0 0 30px grey;

For a softer grey shadow, we can use rgba text-shadow: 0 0 30px rgba(0,0,-0,0.2);

### Spacing

margin

padding



By [raposinha](https://cheatography.com/raposinha/)  
[cheatography.com/raposinha/](https://cheatography.com/raposinha/)

Not published yet.  
Last updated 8th January, 2024.  
Page 2 of 2.

Sponsored by [Readable.com](https://readable.com)  
Measure your website readability!  
<https://readable.com>