

# CSS align-content Property

## Example

Pack lines toward the center of the flex container:

```
div {  
  width: 70px;  
  height: 300px;  
  border: 1px solid #c3c3c3;  
  display: -webkit-flex;  
  display: flex;  
  -webkit-flex-wrap: wrap;  
  flex-wrap: wrap;  
  -webkit-align-content: center;  
  align-content: center;  
}
```

## Definition and Usage

The align-content property modifies the behavior of the flex-wrap property. It is similar to align-items, but instead of aligning flex items, it aligns flex lines.

**Tip:** Use the [justify-content](#) property to align the items on the main-axis (horizontally).

**Note:** There must be multiple lines of items for this property to have any effect.

Default value: stretch

Inherited: no

Animatable: no. [Read about animatable](#)

Version: CSS3

JavaScript syntax: object.style.alignContent="center"

## CSS Syntax

align-content: stretch | center | flex-start | flex-end | space-between | space-around | initial | inherit;

## Property Values

Value	Description	Play it
stretch	Default value. Lines stretch to take up the remaining space	
center	Lines are packed toward the center of the flex container	
flex-start	Lines are packed toward the start of the flex container	

flex-end	Lines are packed toward the end of the flex container
space-between	Lines are evenly distributed in the flex container
space-around	Lines are evenly distributed in the flex container, with half-size spaces on either end
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

## CSS align-items Property

### Example

Center the alignments for all the items of the flexible <div> element:

```
div {
  display: -webkit-flex; /* Safari */
  -webkit-align-items: center; /* Safari 7.0+ */
  display: flex;
  align-items: center;
}
```

### Definition and Usage

The align-items property specifies the default alignment for items inside the flexible container.

**Tip:** Use the align-self property of each item to override the align-items property.

Default value: stretch

Inherited: no

Animatable: no. [Read about animatable](#)

Version: CSS3

JavaScript syntax: object.style.alignItems="center"

### CSS Syntax

align-items: stretch | center | flex-start | flex-end | baseline | initial | inherit;

### Property Values

Value	Description	Play it
stretch	Default. Items are stretched to fit the container	
center	Items are positioned at the center of the container	

flex-start	Items are positioned at the beginning of the container
flex-end	Items are positioned at the end of the container
baseline	Items are positioned at the baseline of the container
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS align-self Property

## Example

Center the alignments for one of the items inside a flexible element:

```
#myBlueDiv {
  -webkit-align-self: center; /* Safari 7.0+ */
  align-self: center;
}
```

## Definition and Usage

The align-self property specifies the alignment for the selected item inside the flexible container.

**Note:** The align-self property overrides the flexible container's align-items property.

Default value:	auto
Inherited:	no
Animatable:	no. <a href="#">Read about animatable</a>
Version:	CSS3

JavaScript syntax: object.style.alignSelf="center"

## CSS Syntax

align-self: auto | stretch | center | flex-start | flex-end | baseline | initial | inherit;

## Property Values

Value	Description
auto	Default. The element inherits its parent container's align-items property, or "stretch" if it has no parent container
stretch	The element is positioned to fit the container
center	The element is positioned at the center of the container

flex-start	The element is positioned at the beginning of the container
flex-end	The element is positioned at the end of the container
baseline	The element is positioned at the baseline of the container
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

# CSS animation Property

## Example

Binding an animation to a <div> element, using the shorthand property:

```
div {
  animation: mymove 5s infinite;
}
```

## Definition and Usage

The animation property is a shorthand property for eight of the animation properties:

- animation-name
- animation-duration
- animation-timing-function
- animation-delay
- animation-iteration-count
- animation-direction
- animation-fill-mode
- animation-play-state

**Note:** Always specify the animation-duration property, otherwise the duration is 0, and will never be played.

Default value: none 0 ease 0 1 normal none running

Inherited: no

Animatable: no. [Read about animatable](#)

Version: CSS3

JavaScript syntax: object.style.animation="mymove 5s infinite"

## CSS Syntax

animation: *name duration timing-function delay iteration-count direction fill-mode play-state*;

## Property Values

Value	Description
<a href="#"><u>animation-name</u></a>	Specifies the name of the keyframe you want to bind to the selector
<a href="#"><u>animation-duration</u></a>	Specifies how many seconds or milliseconds an animation takes to complete
<a href="#"><u>animation-timing-function</u></a>	Specifies the speed curve of the animation
<a href="#"><u>animation-delay</u></a>	Specifies a delay before the animation will start
<a href="#"><u>animation-iteration-count</u></a>	Specifies how many times an animation should be played
<a href="#"><u>animation-direction</u></a>	Specifies whether or not the animation should play in reverse on alternate cycles
<a href="#"><u>animation-fill-mode</u></a>	Specifies what values are applied by the animation outside the time it is executing
<a href="#"><u>animation-play-state</u></a>	Specifies whether the animation is running or paused
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

# CSS animation-delay Property

## Example

Start the animation after 2 seconds:

```
div {  
  animation-delay: 2s;  
}
```

## Definition and Usage

The animation-delay property specifies a delay for the start of an animation.

The animation-delay value is defined in seconds (s) or milliseconds (ms).

Default value: 0s

Inherited: no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.animationDelay="1s"`

### CSS Syntax

`animation-delay: time | initial | inherit;`

### Property Values

Value	Description
<code>time</code>	Optional. Defines the number of seconds (s) or milliseconds (ms) to wait before the animation will start. Default value is 0. Negative values are allowed. If you use negative values, the animation will start as if it had already been playing for N seconds/milliseconds.
<code>initial</code>	Sets this property to its default value. <a href="#">Read about initial</a>
<code>inherit</code>	Inherits this property from its parent element. <a href="#">Read about inherit</a>

---

## CSS animation-direction Property

### Example

Play the animation forwards first, then backwards:

```
div {  
  animation-direction: alternate;  
}
```

### Definition and Usage

The `animation-direction` property defines whether an animation should be played forwards, backwards or in alternate cycles.

**Default value:** normal

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.animationDirection="reverse"`

### CSS Syntax

`animation-direction: normal | reverse | alternate | alternate-reverse | initial | inherit;`

## Property Values

Value	Description	Play it
normal	Default value. The animation is played as normal (forwards)	
reverse	The animation is played in reverse direction (backwards)	
alternate	The animation is played forwards first, then backwards	
alternate-reverse	The animation is played backwards first, then forwards	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS animation-duration Property

## Example

Specify that the animation should complete one cycle in 3 seconds:

```
div {  
  animation-duration: 3s;  
}
```

## Definition and Usage

The animation-duration property defines how long an animation should take to complete one cycle.

<b>Default value:</b>	0
<b>Inherited:</b>	no
<b>Animatable:</b>	no. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS3
<b>JavaScript syntax:</b>	<i>object</i> .style.animationDuration="3s"

## CSS Syntax

animation-duration: *time* | initial | inherit;

## Property Values

Value	Description
-------	-------------

*time* Specifies the length of time an animation should take to complete one cycle. This can be specified in seconds or milliseconds. Default value is 0, which means that no animation will occur

*initial* Sets this property to its default value. [Read about \*initial\*](#)

*inherit* Inherits this property from its parent element. [Read about \*inherit\*](#)

## CSS animation-fill-mode Property

Let the <div> element retain the style values from the last keyframe when the animation ends:

### Definition and Usage

The animation-fill-mode property specifies a style for the element when the animation is not playing (before it starts, after it ends, or both).

CSS animations do not affect the element before the first keyframe is played or after the last keyframe is played. The animation-fill-mode property can override this behavior.

**Default value:** none

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object*.style.animationFillMode="forwards"

### CSS Syntax

animation-fill-mode: none|forwards|backwards|both|initial|inherit;

### Property Values

Value	Description
-------	-------------



none	Default value. Animation will not apply any styles to the element before or after it is executing
forwards	The element will retain the style values that is set by the last keyframe (depends on animation-direction and animation-iteration-count)
backwards	The element will get the style values that is set by the first keyframe (depends on animation-direction), and retain this during the animation-delay period
both	The animation will follow the rules for both forwards and backwards, extending the animation properties in both directions

Let the <div> element get the style values set by the first keyframe before the animation starts (during the animation-delay period):

```
div {
  animation-fill-mode: backwards;
}
```

Let the <div> element get the style values set by the first keyframe before the animation starts, and retain the style values from the last keyframe when the animation ends:

```
div {
  animation-fill-mode: both;
}
```

## CSS backface-visibility Property

### Example

Hide and show the back face of two rotated <div> elements:

```
#div1 {
  backface-visibility: hidden;
}
#div2 {
  backface-visibility: visible;
}
```

### Definition and Usage

The backface-visibility property defines whether or not the back face of an element should be visible when facing the user.

The back face of an element is a mirror image of the front face being displayed.

This property is useful when an element is rotated. It lets you choose if the user should see the back face or not.

**Default value:** visible

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object*.style.backfaceVisibility="hidden"

### CSS Syntax

backface-visibility: visible | hidden | initial | inherit;

### Property Values

Value	Description
visible	Default value. The backside is visible
hidden	The backside is not visible

## CSS background-blend-mode Property

Specify the blending mode to be "lighten":

```
div {  
  width: 400px;  
  height: 400px;  
  background-repeat: no-repeat, repeat;  
  background-image: url("img_tree.gif"), url("paper.gif");  
  background-blend-mode: lighten;  
}
```

### Definition and Usage

The background-blend-mode property defines the blending mode of each background layer (color and/or image).

**Default value:** normal

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object*.style.backgroundBlendMode="screen"

## CSS Syntax

background-blend-mode: normal | multiply | screen | overlay | darken | lighten | color-dodge | saturation | color | luminosity;

## Property Values

Value	Description
normal	This is default. Sets the blending mode to normal
multiply	Sets the blending mode to multiply
screen	Sets the blending mode to screen
overlay	Sets the blending mode to overlay
darken	Sets the blending mode to darken
lighten	Sets the blending mode to lighten
color-dodge	Sets the blending mode to color-dodge
saturation	Sets the blending mode to saturation
color	Sets the blending mode to color
luminosity	Sets the blending mode to luminosity

## Example

Specify the blending mode to be "multiply":

```
div {  
  width: 400px;  
  height: 400px;  
  background-repeat: no-repeat, repeat;  
  background-image: url("img_tree.gif"), url("paper.gif");  
  background-blend-mode: multiply;  
}
```

Specify the blending mode to be "screen":

```
div {  
  width: 400px;  
  height: 400px;
```

```
background-repeat: no-repeat, repeat;
background-image: url("img_tree.gif"), url("paper.gif");
background-blend-mode: screen;
}
```

Example

Specify the blending mode to be "overlay":

```
div {
width: 400px;
height: 400px;
background-repeat: no-repeat, repeat;
background-image: url("img_tree.gif"), url("paper.gif");
background-blend-mode: overlay;
}
```

Example

Specify the blending mode to be "darken":

```
div {
width: 400px;
height: 400px;
background-repeat: no-repeat, repeat;
background-image: url("img_tree.gif"), url("paper.gif");
background-blend-mode: darken;
}
```

Example

Specify the blending mode to be "color-dodge":

```
div {
width: 400px;
height: 400px;
background-repeat: no-repeat, repeat;
background-image: url("img_tree.gif"), url("paper.gif");
background-blend-mode: color-dodge;
}
```

Example

Specify the blending mode to be "saturation":

```
div {
width: 400px;
height: 400px;
background-repeat: no-repeat, repeat;
background-image: url("img_tree.gif"), url("paper.gif");
}
```

```
background-blend-mode: saturation;
}
```

#### Example

Specify the blending mode to be "color":

```
div {
width: 400px;
height: 400px;
background-repeat: no-repeat, repeat;
background-image: url("img_tree.gif"), url("paper.gif");
background-blend-mode: color;
}
```

#### Example

Specify the blending mode to be "luminosity":

```
div {
width: 400px;
height: 400px;
background-repeat: no-repeat, repeat;
background-image: url("img_tree.gif"), url("paper.gif");
background-blend-mode: luminosity;
}
```

#### Example

Specify the blending mode to be "normal":

```
div {
width: 400px;
height: 400px;
background-repeat: no-repeat, repeat;
background-image: url("img_tree.gif"), url("paper.gif");
background-blend-mode: normal;
}
```

## CSS background-clip Property

#### Example

Specify how far the background should extend within an element:

```
div {
border: 10px dotted black;
padding: 15px;
background: lightblue;
```

```
background-clip: padding-box;
}
```

## Definition and Usage

The background-clip property defines how far the background (color or image) should extend within an element.

**Default value:** border-box

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object*.style.backgroundClip="content-box"

## CSS Syntax

background-clip: border-box|padding-box|content-box|initial|inherit;

## Property Values

Value	Description
border-box	Default value. The background extends behind the border
padding-box	The background extends to the inside edge of the border
content-box	The background extends to the edge of the content box

# CSS background-origin Property

## Example

Let the background image start from the upper left corner of the content:

```
#example1 {
border: 10px double black;
padding: 25px;
background: url(paper.gif);
background-repeat: no-repeat;
background-origin: content-box;
}
```

## Definition and Usage

The background-origin property specifies the origin position (the background positioning area) of a background image.

**Note:** This property has no effect if [background-attachment](#) is "fixed".

**Default value:** padding-box

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object*.style.backgroundOrigin="content-box"

## CSS Syntax

background-origin: padding-box | border-box | content-box | initial | inherit;

## Property Values

Value	Description
padding-box	Default value. The background image starts from the upper left corner of the padding edge
border-box	The background image starts from the upper left corner of the border
content-box	The background image starts from the upper left corner of the content

# CSS border-collapse Property

## Example

Set the collapsing borders model for two tables:

```
#table1 {  
  border-collapse: separate;  
}  
  
#table2 {  
  border-collapse: collapse;  
}
```

## Definition and Usage

The border-collapse property sets whether table borders should collapse into a single border or be separated as in standard HTML.

### CSS Syntax

border-collapse: separate|collapse|initial|inherit;

### Property Values

Value	Description
separate	Borders are separated; each cell will display its own borders. This is default.
collapse	Borders are collapsed into a single border when possible (border-spacing and empty-cells properties have no effect)

### Example

When using "border-collapse: separate", the border-spacing property can be used to set the space between the cells:

```
#table1 {  
  border-collapse: separate;  
  border-spacing: 10px;  
}
```

### Example

When using "border-collapse: collapse", the cell that appears first in the code will "win":

```
table, td, th {  
  border: 3px solid red;  
}  
  
#table1 {  
  border-collapse: collapse;  
  border-color: blue;  
}
```

# CSS border-image Property

### Example

Specify an image as the border around an element:



```
#borderimg {  
  border-image: url(border.png) 30 round;  
}
```

## Definition and Usage

The border-image property allows you to specify an image to be used as the border around an element.

The border-image property is a shorthand property for [border-image-source](#), [border-image-slice](#), [border-image-width](#), [border-image-outset](#) and [border-image-repeat](#).

Omitted values are set to their default values.

**Default value:** none 100% 1 0 stretch

## CSS Syntax

border-image: *source slice width outset repeat* | initial | inherit;

## Property Values

Value	Description
<a href="#">border-image-source</a>	The path to the image to be used as a border
<a href="#">border-image-slice</a>	How to slice the border image
<a href="#">border-image-width</a>	The width of the border image
<a href="#">border-image-outset</a>	The amount by which the border image area extends beyond the border box
<a href="#">border-image-repeat</a>	Whether the border image should be repeated, rounded or stretched

# CSS border-image-outset Property

## Example

Place the border image 10 pixels outside the border edges of an element:

```
#borderimg {  
  border-image-source: url(border.png);  
  border-image-outset: 10px;  
}
```

## Definition and Usage

The `border-image-outset` property specifies the amount by which the border image area extends beyond the border box.

**Tip:** Also look at the [border-image](#) property (a shorthand property for setting all the `border-image-*` properties).

## CSS Syntax

`border-image-outset: length | number | initial | inherit;`

**Note:** The `border-image-outset` property can take from one to four values (top, right, bottom, and left sides). If the fourth value is omitted, it is the same as the second. If the third one is also omitted, it is the same as the first. If the second one is also omitted, it is the same as the first.

## Property Values

Value	Description
<i>length</i>	A length unit specifying how far from the edges the border-image will appear. Default value is 0
<i>number</i>	Represent multiples of the corresponding border-width

# CSS border-image-repeat Property

## Example

Specify how to repeat the border image:

```
#borderimg {  
  border-image-source: url(border.png);  
  border-image-repeat: repeat;  
}
```

## Definition and Usage

The `border-image-repeat` property specifies whether the border image should be repeated, rounded or stretched.

**Tip:** Also look at the [border-image](#) property (a shorthand property for setting all the `border-image-*` properties).

## CSS Syntax

`border-image-repeat: stretch | repeat | round | initial | inherit;`

**Note:** This property specifies how the images for the sides and the middle part of the border image are scaled and tiled. So, you can specify two values here. If the second value is omitted, it is assumed to be the same as the first.

### Property Values

Value	Description
stretch	Default value. The image is stretched to fill the area
repeat	The image is tiled (repeated) to fill the area
round	The image is tiled (repeated) to fill the area. If it does not fill the area with a whole number of tiles, the image is rescaled so it fits
space	The image is tiled (repeated) to fill the area. If it does not fill the area with a whole number of tiles, the extra space is distributed around the tiles

## CSS border-image-slice Property

### Example

Specify how to slice the border image:

```
#borderimg {  
  border-image-slice: 30%;  
}
```

### Definition and Usage

The border-image-slice property specifies how to slice the image specified by [border-image-source](#). The image is always sliced into nine sections: four corners, four edges and the middle.

The "middle" part is treated as fully transparent, unless the fill keyword is set.

**Tip:** Also look at the [border-image](#) property (a shorthand property for setting all the border-image-\* properties).

### CSS Syntax

border-image-slice: *number* | % | fill | initial | inherit;

**Note:** The border-image-slice property can take from one to four values. If the fourth value is omitted, it is the same as the second. If the third one is also omitted, it is the same as the first. If the second one is also omitted, it is the same as the first.

## Property Values

Value	Description
<i>number</i>	The number(s) represent pixels for raster images or coordinates for vector images
%	Percentages are relative to the height or width of the image
fill	Causes the middle part of the image to be displayed

# CSS border-image-width Property

## Example

Specify the width of the border image:

```
#borderimg {  
  border-image-source: url(border.png);  
  border-image-width: 10px;  
}
```

## Definition and Usage

The border-image-width property specifies the width of the border image.

**Tip:** Also look at the [border-image](#) property (a shorthand property for setting all the border-image-\* properties).

**Default value:** 1

## CSS Syntax

border-image-width: *number* | % | auto | initial | inherit;

**Note:** The border-image-width property can take from one to four values (top, right, bottom, and left sides). If the fourth value is omitted, it is the same as the second. If the third one is also omitted, it is the same as the first. If the second one is also omitted, it is the same as the first.

## Property Values

Value	Description
<i>length</i>	A length unit (px) specifying the size of the border-width
<i>number</i>	Default value 1. Represents multiples of the corresponding border-width
%	Refers to the size of the border image area: the width of the area for horizontal offsets, the height for vertical offsets

# CSS border-spacing Property

## Example

Set the border-spacing for a table:

```
#table1 {  
  border-collapse: separate;  
  border-spacing: 15px;  
}  
  
#table2 {  
  border-collapse: separate;  
  border-spacing: 15px 50px;  
}
```

## Definition and Usage

The border-spacing property sets the distance between the borders of adjacent cells.

**Note:** This property works only when [border-collapse](#) is separate.

**Default value:** 2px

## CSS Syntax

border-spacing: *length* | initial | inherit;

## Property Values

Value	Description
	Specifies the distance between the borders of adjacent cells in px, cm, etc. Negative values are not allowed.
<i>length length</i>	<ul style="list-style-type: none"><li>If one value is specified, it defines both the horizontal and vertical spacing between cells</li><li>If two values are specified, the first sets the horizontal spacing and the second sets the vertical spacing</li></ul>

# CSS box-decoration-break Property

## Example

Specify the box-decoration-break property:

```
span.ex1 {  
  -webkit-box-decoration-break: clone;  
}
```

```

    -o-box-decoration-break: clone;
    box-decoration-break: clone;
}

span.ex2 {
    -webkit-box-decoration-break: slice;
    -o-box-decoration-break: slice;
    box-decoration-break: slice;
}

```

## Definition and Usage

The `box-decoration-break` property specifies how the background, padding, border, border-image, box-shadow, margin, and clip-path of an element is applied when the box for the element is fragmented.

**Default value:** `slice`

## CSS Syntax

`box-decoration-break: slice|clone|initial|inherit|unset;`

## Property Values

Value	Description
<code>slice</code>	Default. Box decorations are applied to the element as a whole and break at the edges of the element fragments
<code>clone</code>	Box decorations apply to each fragment of the element as if the fragments were individual elements. Borders wrap the four edges of each fragment of the element, and backgrounds are redrawn in full for each fragment

# CSS box-shadow Property

## Example

Add shadows to different `<div>` elements:

```

#example1 {
    box-shadow: 5px 10px;
}

#example2 {
    box-shadow: 5px 10px #888888;
}

```

## Definition and Usage

The box-shadow property attaches one or more shadows to an element.

**Default value:** none

## CSS Syntax

box-shadow: none | *h-offset v-offset blur spread color* | inset | initial | inherit;

**Note:** To attach more than one shadow to an element, add a comma-separated list of shadows (se "Try it Yourself" example below).

## Property Values

Value	Description	Play it
none	Default value. No shadow is displayed	
<i>h-offset</i>	Required. The horizontal offset of the shadow. A positive value puts the shadow on the right side of the box, a negative value puts the shadow on the left side of the box	
<i>v-offset</i>	Required. The vertical offset of the shadow. A positive value puts the shadow below the box, a negative value puts the shadow above the box	
<i>blur</i>	Optional. The blur radius. The higher the number, the more blurred the shadow will be	
<i>spread</i>	Optional. The spread radius. A positive value increases the size of the shadow, a negative value decreases the size of the shadow	
<i>color</i>	Optional. The color of the shadow. The default value is the text color. Look at <a href="#">CSS Color Values</a> for a complete list of possible color values.  <b>Note:</b> In Safari (on PC) the color parameter is required. If you do not specify the color, the shadow is not displayed at all.	
inset	Optional. Changes the shadow from an outer shadow (outset) to an inner shadow	

## Example

Add a blur effect to the shadow:

```
#example1 {  
  box-shadow: 10px 10px 8px #888888;  
}
```

### Example

Define the spread radius of the shadow:

```
#example1 {  
  box-shadow: 10px 10px 8px 10px #888888;  
}
```

### Example

Define multiple shadows:

```
#example1 {  
  box-shadow: 5px 5px blue, 10px 10px red, 15px 15px green;  
}
```

### Example

Add the inset keyword:

```
#example1 {  
  box-shadow: 5px 10px inset;  
}
```

### Example

Images thrown on the table. This example demonstrates how to create "polaroid" pictures and rotate the pictures:

```
div.polaroid {  
  width: 284px;  
  padding: 10px 10px 20px 10px;  
  border: 1px solid #BFBFBF;  
  background-color: white;  
  box-shadow: 10px 10px 5px #aaaaaa;  
}
```

## CSS Tables

### 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;  
}
```

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

---

### Collapse Table Borders

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

#### Example

```
table {  
  border-collapse: collapse;  
}
```

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

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

#### Example

```
table {  
  border: 1px solid black;  
}
```

### Table Width and Height

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 50px:

#### Example

```
table {  
  width: 100%;  
}
```

```
th {  
  height: 50px;  
}
```

## 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.

The following example left-aligns the text in `<th>` elements:

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:

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:

Example

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

## Horizontal Dividers

First Name Last Name Savings		
Peter	Griffin	\$100
Lois	Griffin	\$150

### First Name Last Name Savings

Joe	Swanson	\$300
-----	---------	-------

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

Example

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

### Hoverable Table

Use the `:hover` selector on `<tr>` to highlight table rows on mouse over:

### First Name Last Name Savings

Peter	Griffin	\$100
Lois	Griffin	\$150
Joe	Swanson	\$300

Example

```
tr:hover {background-color: #f5f5f5;}
```

### 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

```
tr:nth-child(even) {background-color: #f2f2f2;}
```

### 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: #4CAF50;  
  color: white;  
}
```

## Responsive Table

A responsive table will display a horizontal scroll bar if the screen is too small to display the full content:

First Name	Last Name	Points	Points	Points	Points	Points	Points	Points	Points	Points	Points	Points	Points
Jill	Smith	50	50	50	50	50	50	50	50	50	50	50	50
Eve	Jackson	94	94	94	94	94	94	94	94	94	94	94	94
Adam	Johnson	67	67	67	67	67	67	67	67	67	67	67	67

Add a container element (like `<div>`) with `overflow-x:auto` around the `<table>` element to make it responsive:

Example

```
<div style="overflow-x:auto;">
```

```
<table>
```

```
... table content ...
```

```
</table>
```

```
</div>
```

## CSS caret-color Property

Set the color of the cursor in input elements:

```
input {  
  caret-color: red;  
}
```

CSS can generate a bunch of different mouse cursors:

```
.alias {cursor: alias;}
.all-scroll {cursor: all-scroll;}
.auto {cursor: auto;}
.cell {cursor: cell;}
.context-menu {cursor: context-menu;}
.col-resize {cursor: col-resize;}
.copy {cursor: copy;}
.crosshair {cursor: crosshair;}
.default {cursor: default;}
.e-resize {cursor: e-resize;}
.ew-resize {cursor: ew-resize;}
.grab {cursor: grab;}
.grabbing {cursor: grabbing;}
.help {cursor: help;}
.move {cursor: move;}
.n-resize {cursor: n-resize;}
.ne-resize {cursor: ne-resize;}
.nesw-resize {cursor: nesw-resize;}
.ns-resize {cursor: ns-resize;}
.nw-resize {cursor: nw-resize;}
.nwse-resize {cursor: nwse-resize;}
.no-drop {cursor: no-drop;}
.none {cursor: none;}
.not-allowed {cursor: not-allowed;}
.pointer {cursor: pointer;}
.progress {cursor: progress;}
.row-resize {cursor: row-resize;}
.s-resize {cursor: s-resize;}
.se-resize {cursor: se-resize;}
.sw-resize {cursor: sw-resize;}
.text {cursor: text;}
.url {cursor: url(myBall.cur),auto;}
.w-resize {cursor: w-resize;}
.wait {cursor: wait;}
.zoom-in {cursor: zoom-in;}
.zoom-out {cursor: zoom-out;}
```

## CSS direction Property

### Example

Set the text direction to "right-to-left":

```
p.rtl {
  direction: rtl;
}
```

# CSS filter Property

## Example

Change all images to black and white (100% gray):

```
img {  
  -webkit-filter: grayscale(100%); /* Safari 6.0 - 9.0 */  
  filter: grayscale(100%);  
}
```

## Definition and Usage

The filter property defines visual effects (like blur and saturation) to an element (often `<img>`).

**Default value:** none

## CSS Syntax

filter: none | blur() | brightness() | contrast() | drop-shadow() | grayscale() | hue-rotate() | invert() | opacity() | saturate() | sepia() | url();

## Filter Functions

**Note:** The filters that use percentage values (i.e. 75%), also accept the value as decimal (i.e. 0.75).

Filter	Description	Play it
none	Default value. Specifies no effects	
blur(px)	Applies a blur effect to the image. A larger value will create more blur.  If no value is specified, 0 is used.	
brightness(%)	Adjusts the brightness of the image.  0% will make the image completely black. 100% (1) is default and represents the original image. Values over 100% will provide brighter results.	
contrast(%)	Adjusts the contrast of the image.  0% will make the image completely black. 100% (1) is default and represents the	

original image.

Values over 100% will provide results with less contrast.

Applies a drop shadow effect to the image.

**Possible values:**

*h-shadow* - Required. Specifies a pixel value for the horizontal shadow. Negative values place the shadow to the left of the image.

*v-shadow* - Required. Specifies a pixel value for the vertical shadow. Negative values place the shadow above the image.

*blur* - Optional. This is the third value, and must be in pixels. Adds a blur effect to the shadow. A larger value will create more blur (the shadow becomes bigger and lighter). Negative values are not allowed. If no value is specified, 0 is used (the shadow's edge is sharp).

`drop-shadow(h-shadow v-shadow blur spread color)`

*spread* - Optional. This is the fourth value, and must be in pixels. Positive values will cause the shadow to expand and grow bigger, and negative values will cause the shadow to shrink. If not specified, it will be 0 (the shadow will be the same size as the element).

**Note:** Chrome, Safari and Opera, and maybe other browsers, do not support this 4th length; it will not render if added.

*color* - Optional. Adds a color to the shadow. If not specified, the color depends on the browser (often black).

An example of creating a red shadow, which is 8px big both horizontally and vertically, with a blur effect of 10px:

`filter: drop-shadow(8px 8px 10px red);`

**Tip:** This filter is similar to the [box-shadow](#) property.

	Converts the image to grayscale.
grayscale(%)	<p>0% (0) is default and represents the original image. 100% will make the image completely gray (used for black and white images).</p> <p><b>Note:</b> Negative values are not allowed.</p>
hue-rotate(deg)	<p>Applies a hue rotation on the image. The value defines the number of degrees around the color circle the image samples will be adjusted. 0deg is default, and represents the original image.</p> <p><b>Note:</b> Maximum value is 360deg.</p>
invert(%)	<p>Inverts the samples in the image.</p> <p>0% (0) is default and represents the original image. 100% will make the image completely inverted.</p> <p><b>Note:</b> Negative values are not allowed.</p>
opacity(%)	<p>Sets the opacity level for the image. The opacity-level describes the transparency-level, where:</p> <p>0% is completely transparent. 100% (1) is default and represents the original image (no transparency).</p> <p><b>Note:</b> Negative values are not allowed. <b>Tip:</b> This filter is similar to the <a href="#">opacity</a> property.</p>
saturate(%)	<p>Saturates the image.</p> <p>0% (0) will make the image completely un-saturated. 100% is default and represents the original image. Values over 100% provides super-saturated results.</p> <p><b>Note:</b> Negative values are not allowed.</p>



Converts the image to sepia.

sepia(%)

0% (0) is default and represents the original image.

100% will make the image completely sepia.

**Note:** Negative values are not allowed.

The url() function takes the location of an XML file that specifies an SVG filter, and may include an anchor to a specific filter element. Example:

url()  
filter: url(svg-url#element-id)

initial Sets this property to its default value. [Read about initial](#)

inherit Inherits this property from its parent element. [Read about inherit](#)

### Blur Example

Apply a blur effect to the image:

```
img {  
  -webkit-filter: blur(5px); /* Safari */  
  filter: blur(5px);  
}
```

### Blur Example 2

Apply a blurred background image:

```
img.background {  
  -webkit-filter: blur(35px); /* Safari */  
  filter: blur(35px);  
}
```

### Brightness Example

Adjust the brightness of the image:

```
img {  
  -webkit-filter: brightness(200%); /* Safari */  
  filter: brightness(200%);  
}
```

### Contrast Example

Adjust the contrast of the image:

```
img {  
  -webkit-filter: contrast(200%); /* Safari */  
}
```

```
filter: contrast(200%);  
}
```

### Drop Shadow Example

Apply a drop shadow effect to the image:

```
img {  
  -webkit-filter: drop-shadow(8px 8px 10px gray); /* Safari */  
  filter: drop-shadow(8px 8px 10px gray);  
}
```

### Grayscale Example

Convert the image to grayscale:

```
img {  
  -webkit-filter: grayscale(50%); /* Safari */  
  filter: grayscale(50%);  
}
```

## CSS font-size-adjust Property

### Example

By specifying the font-size-adjust property, the browser will adjust the font size to be the same regardless of the font family ("verdana" has the aspect value 0.58)

```
div {  
  font-size-adjust: 0.58;  
}
```

### Definition and Usage

The font-size-adjust property gives you better control of the font size when the first selected font is not available.

When a font is not available, the browser uses the second specified font. This could result in a big change for the font size. To prevent this, use the font-size-adjust property.

All fonts have an "aspect value" which is the size-difference between the lowercase letter "x" and the uppercase letter "X".

When the browser knows the "aspect value" for the first selected font, the browser can figure out what font-size to use when displaying text with the second choice font.

**Default value:** none

Inherited: yes

## CSS font-style Property

### Example

Set different font styles for three paragraphs:

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

## CSS font-variant Property

### Example

Set a paragraph to a small-caps font:

```
p.small {  
  font-variant: small-caps;  
}
```

### Definition and Usage

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.

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

## CSS font-weight Property

### Example

Set different font weight for three paragraphs:

```
p.normal {  
  font-weight: normal;  
}
```

```
p.thick {  
  font-weight: bold;  
}
```

```
p.thicker {  
  font-weight: 900;  
}
```

### Definition and Usage

The font-weight property sets how thick or thin characters in text should be displayed.

## CSS @keyframes Rule

### Example

Make an element move gradually 200px down:

```
@keyframes mymove {  
  from {top: 0px;}  
  to {top: 200px;}  
}
```

### Definition and Usage

The @keyframes rule specifies the animation code.

The animation is created by gradually changing from one set of CSS styles to another.

During the animation, you can change the set of CSS styles many times.

Specify when the style change will happen in percent, or with the keywords "from" and "to", which is the same as 0% and 100%. 0% is the beginning of the animation, 100% is when the animation is complete.

**Tip:** For best browser support, you should always define both the 0% and the 100% selectors.

**Note:** Use the animation properties to control the appearance of the animation, and also to bind the animation to selectors.

## Property Values

Value	Description
<i>animationname</i>	Required. Defines the name of the animation.
	Required. Percentage of the animation duration.
<i>keyframes-selector</i>	Legal values: 0-100% from (same as 0%) to (same as 100%)
	<b>Note:</b> You can have many keyframes-selectors in one animation.
<i>css-styles</i>	Required. One or more legal CSS style properties

### Example

Add many keyframe selectors in one animation:

```
@keyframes mymove {  
  0% {top: 0px;}  
  25% {top: 200px;}  
  50% {top: 100px;}  
  75% {top: 200px;}  
  100% {top: 0px;}  
}
```

# CSS letter-spacing Property

### Example

Set the letter spacing for <h1> and <h2> elements:

```
h1 {  
  letter-spacing: 2px;  
}  
  
h2 {  
  letter-spacing: -3px;  
}
```

### Definition and Usage

The letter-spacing property increases or decreases the space between characters in a text.

# CSS line-height Property

## Example

Set the line height in percent:

```
p.small {  
  line-height: 90%;  
}  
  
p.big {  
  line-height: 200%;  
}
```

## Definition and Usage

The line-height property specifies the line height.

**Note:** Negative values are not allowed.

# CSS list-style Property

## Example

Specify all the list properties in one declaration:

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

## Definition and Usage

The list-style shorthand property sets all the list properties in one declaration.

The properties that can be set, are (in order): list-style-type, list-style-position, list-style-image.

If one of the values above are missing, e.g. "list-style:circle inside;", the default value for the missing property will be inserted, if any.

<b>Default value:</b>	disc outside none
-----------------------	-------------------

# CSS list-style-image Property

## Example

Specify an image as the list-item marker in a list:

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

## Definition and Usage

The list-style-image property replaces the list-item marker with an image.

**Note:** Always specify the list-style-type property in addition. This property is used if the image for some reason is unavailable.

**Default value:** none

# CSS list-style-position Property

## Example

Specify that the list-item markers should appear inside the content flow (results in an extra indentation):

```
ul {  
  list-style-position: inside;  
}
```

## Definition and Usage

The list-style-position property specifies if the list-item markers should appear inside or outside the content flow.

**Default value:** outside

Outside:

- Coffee
- Tea
- Coca-cola

Inside:

- Coffee
- Tea
- Coca-cola

# CSS list-style-type Property

## Example

Set some different list styles:

```
ul.circle {list-style-type: circle;}
ul.square {list-style-type: square;}
ol.upper-roman {list-style-type: upper-roman;}
ol.lower-alpha {list-style-type: lower-alpha;}
```

## Definition and Usage

The list-style-type specifies the type of list-item marker in a list.

**Default value:** disc

Value	Description	Play it
disc	Default value. The marker is a filled circle	
armenian	The marker is traditional Armenian numbering	
circle	The marker is a circle	
cjk-ideographic	The marker is plain ideographic numbers	
decimal	The marker is a number	
decimal-leading-zero	The marker is a number with leading zeros (01, 02, 03, etc.)	
georgian	The marker is traditional Georgian numbering	
hebrew	The marker is traditional Hebrew numbering	



hiragana	The marker is traditional Hiragana numbering
hiragana-iroha	The marker is traditional Hiragana iroha numbering
katakana	The marker is traditional Katakana numbering
katakana-iroha	The marker is traditional Katakana iroha numbering
lower-alpha	The marker is lower-alpha (a, b, c, d, e, etc.)
lower-greek	The marker is lower-greek
lower-latin	The marker is lower-latin (a, b, c, d, e, etc.)
lower-roman	The marker is lower-roman (i, ii, iii, iv, v, etc.)
none	No marker is shown
square	The marker is a square
upper-alpha	The marker is upper-alpha (A, B, C, D, E, etc.)
upper-greek	The marker is upper-greek
upper-latin	The marker is upper-latin (A, B, C, D, E, etc.)
upper-roman	The marker is upper-roman (I, II, III, IV, V, etc.)
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

## CSS @media Rule

### Example

Change the background-color if the viewport is 480 pixels wide or wider:

```
@media screen and (min-width: 480px) {
  body {
    background-color: lightgreen;
  }
}
```

### Definition and Usage

The @media rule is used to define different style rules for different media types/devices.

In CSS2 this was called media types, while in CSS3 it is called media queries.

Media queries look at the capability of the device, and can be used to check many things, such as:

- width and height of the viewport
- width and height of the device
- orientation (is the tablet/phone in landscape or portrait mode?)
- resolution
- and much more

### CSS Syntax

```
@media not|only mediatype and (media feature) {  
    CSS-Code;  
}
```

You can also have different *stylesheets* for different media:

```
<link rel="stylesheet" media="mediatype and|not|only (media feature)" href="mystylesheet.css">
```

### Media Types

Value	Description
all	Used for all media type devices
print	Used for printers
screen	Used for computer screens, tablets, smart-phones etc.
speech	Used for screenreaders that "reads" the page out loud
aural	Deprecated. Used for speech and sound synthesizers
braille	Deprecated. Used for braille tactile feedback devices
embossed	Deprecated. Used for paged braille printers
handheld	Deprecated. Used for small or handheld devices
projection	Deprecated. Used for projected presentations, like slides
tty	Deprecated. Used for media using a fixed-pitch character grid, like teletypes and terminals
tv	Deprecated. Used for television-type devices

### Media Features

Value	Description
any-hover	Does any available input mechanism allow the user to hover over elements? (added in Media Queries Level 4)

any-pointer	Is any available input mechanism a pointing device, and if so, how accurate is it? (added in Media Queries Level 4)
aspect-ratio	The ratio between the width and the height of the viewport
color	The number of bits per color component for the output device
color-index	The number of colors the device can display
device-aspect-ratio	Deprecated. The ratio between the width and the height of the device
device-height	Deprecated. The height of the device, such as a computer screen
device-width	Deprecated. The width of the device, such as a computer screen
grid	Whether the device is a grid or bitmap
height	The viewport height
hover	Does the primary input mechanism allow the user to hover over elements? (added in Media Queries Level 4)
inverted-colors	Is the browser or underlying OS inverting colors? (added in Media Queries Level 4)
light-level	Current ambient light level (added in Media Queries Level 4)
max-aspect-ratio	The maximum ratio between the width and the height of the display area
max-color	The maximum number of bits per color component for the output device
max-color-index	The maximum number of colors the device can display
max-device-aspect-ratio	The maximum ratio between the width and the height of the device
max-device-height	The maximum height of the device, such as a computer screen
max-device-width	The maximum width of the device, such as a computer screen
max-height	The maximum height of the display area, such as a browser window
max-monochrome	The maximum number of bits per "color" on a monochrome (greyscale) device
max-resolution	The maximum resolution of the device, using dpi or dpcm
max-width	The maximum width of the display area, such as a browser window
min-aspect-ratio	The minimum ratio between the width and the height of the display area
min-color	The minimum number of bits per color component for the output device
min-color-index	The minimum number of colors the device can display

min-device-aspect-ratio	The minimum ratio between the width and the height of the device
min-device-width	The minimum width of the device, such as a computer screen
min-device-height	The minimum height of the device, such as a computer screen
min-height	The minimum height of the display area, such as a browser window
min-monochrome	The minimum number of bits per "color" on a monochrome (greyscale) device
min-resolution	The minimum resolution of the device, using dpi or dpcm
min-width	The minimum width of the display area, such as a browser window
monochrome	The number of bits per "color" on a monochrome (greyscale) device
orientation	The orientation of the viewport (landscape or portrait mode)
overflow-block	How does the output device handle content that overflows the viewport along the block axis (added in Media Queries Level 4)
overflow-inline	Can content that overflows the viewport along the inline axis be scrolled (added in Media Queries Level 4)
pointer	Is the primary input mechanism a pointing device, and if so, how accurate is it? (added in Media Queries Level 4)
resolution	The resolution of the output device, using dpi or dpcm
scan	The scanning process of the output device
scripting	Is scripting (e.g. JavaScript) available? (added in Media Queries Level 4)
update-frequency	How quickly can the output device modify the appearance of the content (added in Media Queries Level 4)
width	The viewport width

## More Examples

### Example

Use the @media rule to make responsive design:

```
@media only screen and (max-width: 500px) {
  .gridmenu {
    width:100%;
  }

  .gridmain {
    width:100%;
  }
}
```

```
}  
  
.gridright {  
  width:100%;  
}  
}
```

# CSS min-height Property

## Example

Set the minimum height of a <p> element:

```
p {  
  min-height: 100px;  
}
```

## Definition and Usage

The min-height property is used to set the minimum height of an element.

This prevents the value of the height property from becoming smaller than min-height.

**Note:** The value of the min-height property overrides both [max-height](#) and [height](#).

**Default value:** 0

## CSS Syntax

min-height: *length* | initial | inherit;

## Property Values

Value	Description	Play it
<i>length</i>	Default value is 0. Defines the minimum height in px, cm, etc.	
%	Defines the minimum height in percent of the containing block	
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>	
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>	

# CSS min-width Property

## Example

Set the minimum width of a <p> element:

```
p {  
  min-width: 150px;  
}
```

## Definition and Usage

The min-width property is used to set the minimum width of an element.

This prevents the value of the width property from becoming smaller than min-width.

**Note:** The value of the min-width property overrides both [max-width](#) and [width](#).

**Default  
value:** 0

## CSS Syntax

min-width: *length* | initial | inherit;

## Property Values

Value	Description
<i>length</i>	Default value is 0. Defines the minimum width in px, cm, etc.
%	Defines the minimum width in percent of the containing block
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS max-width Property

## Example

Set the maximum width of a <p> element:

```
p {  
  max-width: 100px;  
}
```

## Definition and Usage

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

This prevents the value of the width property from becoming larger than max-width.

**Note:** The value of the max-width property overrides [width](#).

**Default value:** none

## CSS Syntax

max-width: none | *length* | initial | inherit;

## Property Values

Value	Description
none	No maximum width. This is default
<i>length</i>	Defines the maximum width in px, cm, etc.
%	Defines the maximum width in percent of the containing block
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS max-height Property

## Example

Set the maximum height of a <p> element:

```
p {  
  max-height: 50px;  
}
```

## Definition and Usage

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

This prevents the value of the height property from becoming larger than max-height.

**Note:** The value of the max-height property overrides [height](#).

**Default value:** none

## CSS Syntax

max-height: none|*length*|initial|inherit;

## Property Values

Value	Description	Play it
none	No maximum height. This is default	
<i>length</i>	Defines the maximum height in px, cm, etc.	
%	Defines the maximum height in percent of the containing block	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS object-fit Property

## Example

Cut off the sides of an image, preserving the aspect ratio, and fill in the space:

```
img {  
  width: 200px;  
  height: 400px;  
  object-fit: cover;  
}
```

## Definition and Usage

The CSS `object-fit` property is used to specify how an `<img>` or `<video>` should be resized to fit its container.

This property tells the content to fill the container in a variety of ways; such as "preserve that aspect ratio" or "stretch up and take up as much space as possible".

**Default value:** *see individual properties*

## CSS Syntax

object-fit: fill|contain|cover|scale-down|none|initial|inherit;

## Property Values

Value	Description	Try it
fill	This is default. The replaced content is sized to fill the element's content box. If necessary, the object will be stretched or squished to fit	



contain	The replaced content is scaled to maintain its aspect ratio while fitting within the element's content box
cover	The replaced content is sized to maintain its aspect ratio while filling the element's entire content box. The object will be clipped to fit
none	The replaced content is not resized
scale-down	The content is sized as if none or contain were specified (would result in a smaller concrete object size)

initial Sets this property to its default value. [Read about \*initial\*](#)

inherit Inherits this property from its parent element. [Read about \*inherit\*](#)

## CSS opacity Property

### Example

Set the opacity level for a <div> element:

```
div {
  opacity: 0.5;
}
```

More "Try it Yourself" examples below.

---

### Definition and Usage

The opacity property sets the opacity level for an element.

The opacity-level describes the transparency-level, where 1 is not transparent at all, 0.5 is 50% see-through, and 0 is completely transparent.

**Note:** 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. If you do not want to apply opacity to child elements, use RGBA color values instead (See "More Examples" below).

**Default value:** 1

### CSS Syntax

opacity: *number*|initial|inherit;

## Property Values

Value	Description	Play it
<i>number</i>	Specifies the opacity. From 0.0 (fully transparent) to 1.0 (fully opaque)	
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>	
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>	

# CSS order Property

## Example

Set the order of the flexible items:

```
/* Code for Safari 6.1 to 8.0 */
div#myRedDIV {-webkit-order: 2;}
div#myBlueDIV {-webkit-order: 4;}
div#myGreenDIV {-webkit-order: 3;}
div#myPinkDIV {-webkit-order: 1;}
```

```
/* Standard syntax */
div#myRedDIV {order: 2;}
div#myBlueDIV {order: 4;}
div#myGreenDIV {order: 3;}
div#myPinkDIV {order: 1;}
```

## Definition and Usage

The order property specifies the order of a flexible item relative to the rest of the flexible items inside the same container.

**Note:** If the element is not a flexible item, the order property has no effect.

**Default value:** 0

**Inherited:** no

**Animatable:** yes, see *individual properties*. [Read about \*animatable\*](#)

**Version:** CSS3

**JavaScript syntax:** *object*.style.order="2"

## Browser Support

The numbers in the table specify the first browser version that fully supports the property.

Numbers followed by -webkit- or -moz- specify the first version that worked with a prefix.

## CSS Syntax

order: *number* | initial | inherit;

## Property Values

Value	Description
<i>number</i>	Default value 0. Specifies the order for the flexible item
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

# CSS outline Property

## Example

Set the outline around a <p> element:

```
p {  
  outline: #00FF00 dotted thick;  
}
```

## Definition and Usage

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

The outline property is a shorthand property for setting the following individual outline properties:

- outline-width
- outline-style (required)
- outline-color

The outline property is specified as one, two, or three values from the list above. The order of the values does not matter.

**Note:** The outline is not a part of the element's dimensions, therefore the element's width and height properties do not contain the width of the outline.

**Default value:** medium invert none

**Inherited:** no

**Animatable:** yes, *see individual properties*. [Read about animatable](#)

**Version:** CSS2

**JavaScript syntax:** `object.style.outline="#0000FF dotted 5px"`

## CSS Syntax

outline: *outline-width outline-style outline-color* | initial | inherit;

## Property Values

Value	Description
<a href="#">outline-width</a>	Specifies the width of outline
<a href="#">outline-style</a>	Specifies the style of the outline
<a href="#">outline-color</a>	Specifies the color of the outline
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS outline-color Property

## Example

Set the color of a dotted outline:

```
p {  
  outline-style: dotted;  
  outline-color: #00ff00;  
}
```

## Definition and Usage

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

The outline-color property specifies the color of an outline.

**Note:** Always declare the outline-style property before the outline-color property. An element must have an outline before you change the color of it.

**Default value:** invert

**Inherited:** no

**Animatable:** yes, see *individual properties*. [Read about animatable](#)

**Version:** CSS2

**JavaScript syntax:** `object.style.outlineColor="#FF0000"`

## Tips and Notes

An outline is a line around an element. It is displayed around the margin of the element. However, it is different from the border property.

The outline is not a part of the element's dimensions, therefore the element's width and height properties do not contain the width of the outline.

## CSS Syntax

`outline-color: invert|color|initial|inherit;`

## Property Values

Value	Description	Play it
invert	Performs a color inversion. This ensures that the outline is visible, regardless of color background. This is default	
<i>color</i>	Specifies the color of the outline. Look at <a href="#">CSS Color Values</a> for a complete list of possible color values.	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS outline-offset Property

## Example

Specify an outline 15px outside the border edge:

```
div {  
  border: 2px solid black;  
  outline: 2px solid red;  
  outline-offset: 15px;  
}
```

## Definition and Usage

The outline-offset property adds space between an outline and the edge or border of an element.

The space between an element and its outline is transparent.

Outlines differ from borders in three ways:

- An outline is a line drawn around elements, outside the border edge
- An outline does not take up space
- An outline may be non-rectangular

**Default value:** 0

**Inherited:** no

**Animatable:** yes, see individual properties. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object.style.outlineOffset="15px"*

## CSS Syntax

outline-offset: *length* | initial | inherit;

## Property Values

Value	Description
<i>length</i>	The distance the outline is outset from the border edge. Default value is 0
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS outline-style Property

## Example

Set the style of an outline:

```
p {  
  outline-style: dotted;  
}
```

## Definition and Usage

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

The outline-style property specifies the style of an outline.

<b>Default value:</b>	none
<b>Inherited:</b>	no
<b>Animatable:</b>	no. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS2
<b>JavaScript syntax:</b>	<i>object.style.outlineStyle="dashed"</i>

## Tips and Notes

An outline is a line around an element. It is displayed around the margin of the element. However, it is different from the border property.

The outline is not a part of the element's dimensions, therefore the element's width and height properties do not contain the width of the outline.

**Note:** IE8 supports the outline-style property only if a !DOCTYPE is specified.

## CSS Syntax

outline-style: none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | initial | inherit;

## Property Values

Value	Description	Play it
none	Specifies no outline. This is default	
hidden	Specifies a hidden outline	
dotted	Specifies a dotted outline	
dashed	Specifies a dashed outline	
solid	Specifies a solid outline	
double	Specifies a double outliner	
groove	Specifies a 3D grooved outline. The effect depends on the outline-color value	

ridge	Specifies a 3D ridged outline. The effect depends on the outline-color value
inset	Specifies a 3D inset outline. The effect depends on the outline-color value
outset	Specifies a 3D outset outline. The effect depends on the outline-color value
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

# CSS overflow Property

## Example

Set the overflow property to scroll:

```
div {
  width: 150px;
  height: 150px;
  overflow: scroll;
}
```

## Definition and Usage

The overflow property specifies what happens if content overflows an element's box.

This property specifies whether to clip content or to add scrollbars when an element's content is too big to fit in a specified area.

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

**Default value:** visible

**Inherited:** no

**Animatable:** no. [Read about \*animatable\*](#)

**Version:** CSS2

**JavaScript syntax:** `object.style.overflow="scroll"`

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



## Property Values

Value	Description	Play it
visible	The overflow is not clipped. It renders outside the element's box. This is default	
hidden	The overflow is clipped, and the rest of the content will be invisible	
scroll	The overflow is clipped, but a scroll-bar is added to see the rest of the content	
auto	If overflow is clipped, a scroll-bar should be added to see the rest of the content	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS overflow-x Property

## Example

Clip the left/right edges of the content inside the <div> element - if it overflows the element's content area:

```
div {  
  overflow-x: hidden;  
}
```

## Definition and Usage

The overflow-x property specifies what to do with the left/right edges of the content - if it overflows the element's content area.

**Tip:** Use the [overflow-y](#) property to determine clipping at the top and bottom edges.

**Default value:** visible

**Inherited:** no

**Animatable:** no. [Read about](#) *animatable*

**Version:** CSS3

**JavaScript syntax:** *object.style.overflowX="scroll"*

## CSS Syntax

overflow-x: visible|hidden|scroll|auto|initial|inherit;

## Property Values

Value	Description	Play it
visible	The content is not clipped, and it may be rendered outside the content box	
hidden	The content is clipped - and no scrolling mechanism is provided	
scroll	The content is clipped and a scrolling mechanism is provided	
auto	Should cause a scrolling mechanism to be provided for overflowing boxes	
initial	Sets this property to its default value. <a href="#">Read</a>	

[about](#)  
[initial](#)

inherit Inherits this property from its parent element.  
[Read about inherit](#)

## CSS overflow-y Property

### Example

Clip the top/bottom edges of the content inside the <div> element - if it overflows the element's content area:

```
div {  
  overflow-y: hidden;  
}
```

### Definition and Usage

The overflow-y property specifies what to do with the top/bottom edges of the content - if it overflows the element's content area.

**Tip:** Use the [overflow-x](#) property to determine clipping at the left and right edges.

**Default value:** visible

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object.style.overflowY="scroll"*

### CSS Syntax

overflow-y: visible | hidden | scroll | auto | initial | inherit;

### Property Values

Value

Description

Play  
it

visible	Default value. The content is not clipped, and it may be rendered outside the content box
hidden	The content is clipped - and no scrolling mechanism is provided
scroll	The content is clipped and a scrolling mechanism is provided
auto	Should cause a scrolling mechanism to be provided for overflowing boxes
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

## CSS perspective Property

## Example

Set the perspective from where an element is viewed:

```
div {  
  -webkit-perspective: 500px; /* Safari 4-8 */  
  perspective: 500px;  
}
```

## Definition and Usage

The perspective property defines how many pixels a 3D element is placed from the view. This property allows you to change the perspective on how 3D elements are viewed.

When defining the perspective property for an element, it is the CHILD elements that get the perspective view, NOT the element itself.

**Note:** The perspective property only affects 3D transformed elements!

**Tip:** Use this property together with the [perspective-origin](#) property, which allows you to change the bottom position of 3D elements.

To better understand the perspective property, [view a demo](#).

**Default value:** none

**Inherited:** no

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.perspective="50px"`

## CSS Syntax

perspective: *length* | none;

## Property Values

Property Value	Description
<i>length</i>	How far the element is placed from the view
none	Default value. Same as 0. The perspective is not set
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS perspective-origin Property

## Example

Set a 3D element's base placement:

```
div {  
  -webkit-perspective: 150px; /* Safari 4-8 */  
  -webkit-perspective-origin: 10% 10%; /* Safari 4-8 */  
  perspective: 150px;  
  perspective-origin: 10% 10%;  
}
```

## Definition and Usage

The perspective-origin property defines where a 3D element is based in the x- and the y-axis. This property allows you to change the bottom position of 3D elements.

When defining the perspective-origin property for an element, it is the CHILD elements that are positioned, NOT the element itself.

**Note:** This property must be used together with the [perspective](#) property, and only affects 3D transformed elements!

To better understand the perspective-origin property, [view a demo](#).

**Default value:** 50% 50%

**Inherited:** no

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.perspectiveOrigin="10px 50%"`

## CSS Syntax

perspective-origin: *x-axis y-axis* | initial | inherit;

## Property Values

### Property Value

### Description

Defining where the view is placed at the x-axis

*x-axis*

Possible values:

- left
- center
- right

- *length*
- *%*

Default value: 50%

Defining where the view is placed at the y-axis

Possible values:

- y-axis*
- top
  - center
  - bottom
  - *length*
  - *%*

Default value: 50%

initial      Sets this property to its default value. [Read about](#) *initial*

inherit      Inherits this property from its parent element. [Read about](#) *inherit*

# CSS position Property

## Example

Position an <h2> element:

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

## Definition and Usage

The position property specifies the type of positioning method used for an element (static, relative, absolute or fixed).

**Default value:** static

**Inherited:** no

**Animatable:** no. [Read about](#) *animatable*

**Version:** CSS2

**JavaScript syntax:** *object.style.position="absolute"*

**Note:** The `sticky` value is not supported in Internet Explorer or Edge 15 and earlier versions.

## CSS Syntax

position: static|absolute|fixed|relative|sticky|initial|inherit;

## Property Values

Value	Description	Play it
static	Default value. Elements render in order, as they appear in the document flow	
absolute	The element is positioned relative to its first positioned (not static) ancestor element	
fixed	The element is positioned relative to the browser window	
relative	The element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position	
	The element is positioned based on the user's scroll position	
sticky	A sticky element toggles between <code>relative</code> and <code>fixed</code> , depending on the scroll position. It is positioned relative until a given offset position is met in the viewport - then it "sticks" in place (like <code>position:fixed</code> ).	
	<b>Note:</b> Not supported in IE/Edge 15 or earlier. Supported in Safari from version 6.1 with a <code>-webkit-</code> prefix.	
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>	
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>	

# CSS quotes Property

## Example

Specify the quotation marks for quotations:

```
q {  
  quotes: "\00ab" "\00bb";  
}
```

## Definition and Usage

The quotes property sets the type of quotation marks for quotations.

**Default value:** *not specified*

**Inherited:** yes



**Animatable:** no. [Read about](#) *animatable*

**Version:** CSS2

**JavaScript syntax:** *object.style.quotes=""\253' '\273""*

## CSS Syntax

quotes: none | *string* | initial | inherit;

## Property Values

Value	Description	Play it
none	Specifies that the "open-quote" and "close-quote" values of the "content" property will not produce any quotation marks	
<i>string string string string</i>	Specifies which quotation marks to use. The first two values specifies the first level of quotation embedding, the next two values specifies the next level of quote embedding, etc	
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>	
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>	

## Quotation Mark Characters

Result	Description	Entity Number
"	double quote	\0022
'	single quote	\0027
‹	single, left angle quote	\2039
›	single, right angle quote	\203A
«	double, left angle quote	\00AB
»	double, right angle quote	\00BB
‘	left quote (single high-6)	\2018
’	right quote (single high-9)	\2019
“	left quote (double high-6)	\201C
”	right quote (double high-9)	\201D

„ double quote (double low-9) \201E

# CSS `resize` Property

## Example

Specify that a `<div>` element should be resizable by the user:

```
div {  
  resize: both;  
  overflow: auto;  
}
```

## Definition and Usage

The `resize` property specifies whether or not an element is resizable by the user.

**Note:** The `resize` property applies to elements whose computed overflow value is something other than "visible".

**Default value:** none

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.resize="both"`

## CSS Syntax

`resize: none|both|horizontal|vertical|initial|inherit;`

## Property Values

Value	Description	Play it
none	Default value. The user cannot resize the element	
both	The user can adjust both the height and the width of the element	
horizontal	The user can adjust the width of the element	
vertical	The user can adjust the height of the element	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS tab-size Property

## Example

Set the tab-size of a <pre> element:

```
pre {-moz-tab-size: 16;} /* Firefox */  
pre {tab-size: 16;}
```

## Definition and Usage

The tab-size property specifies the length of the space used for the tab character.

In HTML, the tab character is usually displayed as a single space-character, except for some elements, like <textarea> and <pre>, and the result of the tab-size property will only be visible for these elements.

**Default value:** 8

**Inherited:** yes

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object.style.tabSize="16"*

**Note:** No browser is currently supporting the value as a length unit.

---

## CSS Syntax

tab-size: *number* | *length* | initial | inherit;

## Property Values

Value	Description	Play it
<i>number</i>	Default value 8. Specifies the number of space-characters that should be displayed for each tab-character	
<i>length</i>	Specifies the length of a tab-character. This property value is not supported in any of the major browsers	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS text-align Property

## Example

Set the text alignment for <h1>, <h2>, and <h3> elements:

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

## Definition and Usage

The text-align property specifies the horizontal alignment of text in an element.

**Default value:** left if direction is ltr, and right if direction is rtl

**Inherited:** yes

**Animatable:** no. [Read about animatable](#)

**Version:** CSS1

**JavaScript syntax:** `object.style.textAlign="right"`

## CSS Syntax

text-align: left | right | center | justify | initial | inherit;

## Property Values

Value	Description	Play it
left	Aligns the text to the left	
right	Aligns the text to the right	
center	Centers the text	
justify	Stretches the lines so that each line has equal width (like in newspapers and magazines)	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	

inherit

Inherits this property from its parent element. [Read about \*inherit\*](#)

# CSS text-align-last Property

## Example

Align the last line of a <p> element to the right:

```
p {  
  text-align: justify;  
  -moz-text-align-last: right; /* Code for Firefox */  
  text-align-last: right;  
}
```

## Definition and Usage

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

**Note:** The text-align-last property will only work for elements with the text-align property set to "justify".

**Default value:** auto

**Inherited:** yes

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object.style.textAlignLast="right"*

## CSS Syntax

text-align-last: auto|left|right|center|justify|start|end|initial|inherit;

## Property Values

Value	Description	Play it
auto	Default value. The last line is justified and aligned left	
left	The last line is aligned to the left	
right	The last line is aligned to the right	
center	The last line is center-aligned	
justify	The last line is justified as the rest of the lines	

start	The last line is aligned at the beginning of the line (left if the text-direction is left-to-right, and right if the text-direction is right-to-left)
end	The last line is aligned at the end of the line (right if the text-direction is left-to-right, and left if the text-direction is right-to-left)
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

## CSS text-decoration Property

### Example

Set the text decoration for <h1>, <h2>, and <h3> elements:

```
h1 {
  text-decoration: underline;
}

h2 {
  text-decoration: line-through;
}

h3 {
  text-decoration: underline;
}
```

### Definition and Usage

The text-decoration property specifies the decoration added to text.

**Note:** The text-decoration property is a shorthand property for [text-decoration-line](#), [text-decoration-color](#), and [text-decoration-style](#), but this is currently not supported in any of the major browsers.

**Note:** You can use the [text-decoration-color](#) property to change the color of the decoration, otherwise the color is the same as the color of the text.

**Default value:** none

**Inherited:** no

**Animatable:** no, *see individual properties*. [Read about animatable](#)

**Version:** CSS1, renewed in CSS3

**JavaScript syntax:** `object.style.textDecoration="underline"`

## CSS Syntax

text-decoration: none | underline | overline | line-through | initial | inherit;

## Property Values

Value	Description	Play it
none	Defines a normal text. This is default	
underline	Defines a line below the text	
overline	Defines a line above the text	
line-through	Defines a line through the text	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS text-decoration-color Property

## Example

Change the color of the line, in an underlined text:

```
p {  
  text-decoration: underline;  
  text-decoration-color: red;  
}
```

## Definition and Usage

The text-decoration-color property specifies the color of the text-decoration (underlines, overlines, linethroughs).

**Note:** The text-decoration-color property will only have an effect on elements with a visible text-decoration.

**Default value:** `currentColor`

**Inherited:** no

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.textDecorationColor="red"`

### CSS Syntax

`text-decoration-color: color|initial|inherit;`

### Property Values

Value	Description	Play it
<i>color</i>	Specifies the color of the text-decoration	
<i>initial</i>	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>	
<i>inherit</i>	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>	

## CSS text-decoration-line Property

### Example

Display <p> elements with a line on top:

```
p {  
  text-decoration-line: overline;  
}
```

### Definition and Usage

The text-decoration-line property specifies what type of line, if any, the decoration will have.

**Note:** You can also set the text-decoration-line using the text-decoration property, which is a short-hand property for the text-decoration-line, text-decoration-style, and the text-decoration-color properties.

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

**Default value:** none

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.textDecorationLine="overline"`



## CSS Syntax

text-decoration-line: none|underline|overline|line-through|initial|inherit;

## Property Values

Value	Description	Play it
none	Default value. Specifies no line for the text-decoration	
underline	Specifies that a line will be displayed under the text	
overline	Specifies that a line will be displayed over the text	
line-through	Specifies that a line will be displayed through the text	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS text-decoration-style Property

## Example

Display a wavy line under a <p> element:

```
p {  
  text-decoration-style: wavy;  
  text-decoration: underline;  
}
```

## Definition and Usage

The text-decoration-style property specifies how the line, if any, will display.

**Default value:** solid

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object.style.textDecorationStyle="wavy"*

## CSS Syntax

text-decoration-style: solid|double|dotted|dashed|wavy|initial|inherit;

## Property Values

Value	Description	Play it
solid	Default value. The line will display as a single line	
double	The line will display as a double line	
dotted	The line will display as a dotted line	
dashed	The line will display as a dashed line	
wavy	The line will display as a wavy line	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS text-indent Property

## Example

Indent the first line of all <p> elements with 50 pixels:

```
p {  
  text-indent: 50px;  
}
```

## Definition and Usage

The text-indent property specifies the indentation of the first line in a text-block.

**Note:** Negative values are allowed. The first line will be indented to the left if the value is negative.

**Default value:** 0

**Inherited:** yes

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS1

**JavaScript syntax:** *object.style.textIndent="50px"*

## CSS Syntax

text-indent: *length* | initial | inherit;

## Property Values

Value	Description	Play it
<i>length</i>	Defines a fixed indentation in px, pt, cm, em, etc. Default value is 0	
%	Defines the indentation in % of the width of the parent element	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS text-justify Property

## Example

Justification changes spacing between words:

```
div {  
  text-align: justify;  
  text-justify: inter-word;  
}
```

## Definition and Usage

The text-justify property specifies the justification method to use when text-align is set to "justify".

This property specifies how justified text should be aligned and spaced.






<b>Default value:</b>	auto
<b>Inherited:</b>	yes
<b>Animatable:</b>	no. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS3
<b>JavaScript syntax:</b>	<i>object.style.textJustify="inter-word"</i>

## CSS Syntax

text-justify: auto|inter-word|inter-ideograph|inter-cluster|distribute|kashida|trim|initial|inherit;

## Property Values

Value	Description	Play it
auto	The browser determines the justification algorithm	
inter-word	Increases/Decreases the space between words	
inter-ideograph	Justifies content with ideographic text	
inter-cluster	Only content that does not contain any inter-word spacing (such as Asian languages) is justified	
distribute	Spacing between letters and words are increased or decreased as necessary	
kashida	Justifies content by elongating characters	
trim		
none	The justification is disabled	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

Browser Support					
The numbers in the table specify the first browser version that fully supports the property.					
Property					
text-justify	Not supported	5.5	Not supported	10.0.3	Not supported

# CSS text-overflow Property

## Definition and Usage

The text-overflow property specifies how overflowed content that is not displayed should be signaled to the user.

**Default value:** clip

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.textOverflow="ellipsis"`

### CSS Syntax

`text-overflow: clip|ellipsis|string|initial|inherit;`

### Property Values

Value	Description	Play it
clip	Default value. Clips the text	
ellipsis	Render an ellipsis ("...") to represent clipped text	
<i>string</i>	Render the given string to represent clipped text	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

## Browser Support

The numbers in the table specify the first browser version that fully supports the property.

Numbers followed by -o- specify the first version that worked with a prefix.

Property					
text-overflow	4.0	6.0	7.0	3.1	11.0 9.0 -o-

# CSS text-shadow Property

### Example

Basic text-shadow:

```
h1 {  
  text-shadow: 2px 2px #ff0000;  
}
```

### Definition and Usage

The text-shadow property adds shadow to text.

This property accepts a comma-separated list of shadows to be applied to the text.

**Default value:** *none*

**Inherited:** yes

**Animatable:** yes. [Read about](#) *animatable*

**Version:** CSS3

**JavaScript syntax:** *object.style.textShadow="2px 5px 5px red"*

## CSS Syntax

`text-shadow: h-shadow v-shadow blur-radius color|none|initial|inherit;`

**Note:** To add more than one shadow to the text, add a comma-separated list of shadows.

## Property Values

Value	Description	Play it
<i>h-shadow</i>	Required. The position of the horizontal shadow. Negative values are allowed	
<i>v-shadow</i>	Required. The position of the vertical shadow. Negative values are allowed	
<i>blur-radius</i>	Optional. The blur radius. Default value is 0	
<i>color</i>	Optional. The color of the shadow. Look at <a href="#">CSS Color Values</a> for a complete list of possible color values	
<i>none</i>	Default value. No shadow	
<i>initial</i>	Sets this property to its default value. <a href="#">Read about</a> <i>initial</i>	
<i>inherit</i>	Inherits this property from its parent element. <a href="#">Read about</a> <i>inherit</i>	

# CSS text-transform Property

## Example

Transform text in different <p> elements:

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

```
p.capitalize {  
  text-transform: capitalize;  
}
```

## Definition and Usage

The text-transform property controls the capitalization of text.

**Default value:** none

**Inherited:** yes

**Version:** CSS1

**JavaScript syntax:** *object.style.textTransform="uppercase"*

## CSS Syntax

text-transform: none | capitalize | uppercase | lowercase | initial | inherit;

## Property Values

Value	Description	Play it
none	No capitalization. The text renders as it is. This is default	
capitalize	Transforms the first character of each word to uppercase	
uppercase	Transforms all characters to uppercase	
lowercase	Transforms all characters to lowercase	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	

# CSS transform Property

## Example

Rotate a <div> element:

```
div {  
  transform: rotate(7deg);  
}
```

## Definition and Usage

The transform property applies a 2D or 3D transformation to an element. This property allows you to rotate, scale, move, skew, etc., elements.

To better understand the transform property, [view a demo](#).

**Default value:** none

**Inherited:** no

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.transform="rotate(7deg)"`

## Syntax

`transform: none | transform-functions | initial | inherit;`

## Property Values

Value	Description	Play it
none	Defines that there should be no transformation	
<code>matrix(<i>n,n,n,n,n,n</i>)</code>	Defines a 2D transformation, using a matrix of six values	
<code>matrix3d(<i>n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n</i>)</code>	Defines a 3D transformation, using a 4x4 matrix of 16 values	
<code>translate(<i>x,y</i>)</code>	Defines a 2D translation	
<code>translate3d(<i>x,y,z</i>)</code>	Defines a 3D translation	
<code>translateX(<i>x</i>)</code>	Defines a translation, using only the value for the X-axis	
<code>translateY(<i>y</i>)</code>	Defines a translation, using only the value for the Y-axis	
<code>translateZ(<i>z</i>)</code>	Defines a 3D translation, using only the value for the Z-axis	
<code>scale(<i>x,y</i>)</code>	Defines a 2D scale transformation	
<code>scale3d(<i>x,y,z</i>)</code>	Defines a 3D scale transformation	
<code>scaleX(<i>x</i>)</code>	Defines a scale transformation by giving a value for the X-axis	
<code>scaleY(<i>y</i>)</code>	Defines a scale transformation by giving a value for the Y-axis	



<code>scaleZ(z)</code>	Defines a 3D scale transformation by giving a value for the Z-axis
<code>rotate(<i>angle</i>)</code>	Defines a 2D rotation, the angle is specified in the parameter
<code>rotate3d(x,y,z,<i>angle</i>)</code>	Defines a 3D rotation
<code>rotateX(<i>angle</i>)</code>	Defines a 3D rotation along the X-axis
<code>rotateY(<i>angle</i>)</code>	Defines a 3D rotation along the Y-axis
<code>rotateZ(<i>angle</i>)</code>	Defines a 3D rotation along the Z-axis
<code>skew(x-<i>angle</i>,y-<i>angle</i>)</code>	Defines a 2D skew transformation along the X- and the Y-axis
<code>skewX(<i>angle</i>)</code>	Defines a 2D skew transformation along the X-axis
<code>skewY(<i>angle</i>)</code>	Defines a 2D skew transformation along the Y-axis
<code>perspective(<i>n</i>)</code>	Defines a perspective view for a 3D transformed element
<code>initial</code>	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>
<code>inherit</code>	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>

## CSS transform-origin Property

### Example

Set a rotated element's base placement:

```
div {
  transform: rotate(45deg);
  transform-origin: 20% 40%;
}
```

### Definition and Usage

The transform-origin property allows you to change the position of transformed elements.

2D transformations can change the x- and y-axis of an element. 3D transformations can also change the z-axis of an element.

To better understand the transform-origin property, [view a demo](#).

**Note:** This property must be used together with the [transform](#) property.

**Tip:** To better understand this property for **3D transforms**, [view a demo](#).

**Default value:** 50% 50% 0

**Inherited:** no

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.transformOrigin="0 0"`

## CSS Syntax

`transform-origin: x-axis y-axis z-axis | initial | inherit;`

## Property Values

Property Value	Description
	Defines where the view is placed at the x-axis. Possible values:
<i>x-axis</i>	<ul style="list-style-type: none"><li>• left</li><li>• center</li><li>• right</li><li>• <i>length</i></li><li>• %</li></ul>
	Defines where the view is placed at the y-axis. Possible values:
<i>y-axis</i>	<ul style="list-style-type: none"><li>• top</li><li>• center</li><li>• bottom</li><li>• <i>length</i></li><li>• %</li></ul>
	Defines where the view is placed at the z-axis (for 3D transformations). Possible values:
<i>z-axis</i>	<ul style="list-style-type: none"><li>• <i>length</i></li></ul>
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS transform-style Property

## Example

Let the transformed child elements preserve the 3D transformations:

```
div {  
  transform: rotateY(60deg);  
  transform-style: preserve-3d;  
}
```

## Definition and Usage

The transform-style property specifies how nested elements are rendered in 3D space.

**Note:** This property must be used together with the [transform](#) property.

To better understand the transform-style property, [view a demo](#).

**Default value:** flat

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.transformStyle="preserve-3d"`

## CSS Syntax

transform-style: flat|preserve-3d|initial|inherit;

## Property Values

Property Value	Description
flat	Specifies that child elements will NOT preserve its 3D position. This is default
preserve-3d	Specifies that child elements will preserve its 3D position
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS transition Property

## Example

Hover over a <div> element to gradually change the width from 100px to 300px:

```
div {  
  width: 100px;  
  transition: width 2s;  
}
```

```
div:hover {  
  width: 300px;  
}
```

## Definition and Usage

The transition property is a shorthand property for the four transition properties:

transition-property, transition-duration, transition-timing-function, and transition-delay.

**Note:** Always specify the transition-duration property, otherwise the duration is 0s, and the transition will have no effect.

**Default value:** all 0s ease 0s

**Inherited:** no

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** *object.style.transition="all 2s"*

## Property Values

Value	Description
<a href="#"><u>transition-property</u></a>	Specifies the name of the CSS property the transition effect is for
<a href="#"><u>transition-duration</u></a>	Specifies how many seconds or milliseconds the transition effect takes to complete
<a href="#"><u>transition-timing-function</u></a>	Specifies the speed curve of the transition effect
<a href="#"><u>transition-delay</u></a>	Defines when the transition effect will start
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS transition-delay Property

## Example

Wait 2 seconds before the transition effect starts:

```
div {  
  transition-delay: 2s;  
}
```

## Definition and Usage

The transition-delay property specifies when the transition effect will start.

The transition-delay value is defined in seconds (s) or milliseconds (ms).

<b>Default value:</b>	0s
<b>Inherited:</b>	no
<b>Animatable:</b>	no. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS3
<b>JavaScript syntax:</b>	<i>object.style.transitionDelay="2s"</i>

## CSS Syntax

transition-delay: *time* | initial | inherit;

## Property Values

Value	Description
<i>time</i>	Specifies the number of seconds or milliseconds to wait before the transition effect will start
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS transition-duration Property

## Example

Let the transition effect last for 5 seconds:

```
div {  
  transition-duration: 5s;  
}
```

## Definition and Usage

The transition-duration property specifies how many seconds (s) or milliseconds (ms) a transition effect takes to complete.

<b>Default value:</b>	0s
<b>Inherited:</b>	no
<b>Animatable:</b>	no. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS3
<b>JavaScript syntax:</b>	<code>object.style.transitionDuration="1s"</code>

### CSS Syntax

transition-duration: *time* | initial | inherit;

### Property Values

Value	Description
<i>time</i>	Specifies how many seconds or milliseconds a transition effect takes to complete. Default value is 0s, meaning there will be no effect
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS transition-property Property

### Example

Hover over a <div> element, and change the width with a smooth transition effect:

```
div {
  transition-property: width;
}

div:hover {
  width: 300px;
}
```

### Definition and Usage

The transition-property property specifies the name of the CSS property the transition effect is for (the transition effect will start when the specified CSS property changes).

**Tip:** A transition effect could typically occur when a user hover over an element.

**Note:** Always specify the transition-duration property, otherwise the duration is 0, and the transition will have no effect.

<b>Default value:</b>	all
<b>Inherited:</b>	no
<b>Animatable:</b>	no. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS3
<b>JavaScript syntax:</b>	<code>object.style.transitionProperty="width,height"</code>

### CSS Syntax

transition-property: none|all|*property*|initial|inherit;

### Property Values

Value	Description
none	No property will get a transition effect
all	Default value. All properties will get a transition effect
<i>property</i>	Defines a comma separated list of CSS property names the transition effect is for
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

## CSS transition-timing-function Property

### Example

A transition effect with the same speed from start to end:

```
div {
  transition-timing-function: linear;
}
```

### Definition and Usage

The transition-timing-function property specifies the speed curve of the transition effect.

This property allows a transition effect to change speed over its duration.

<b>Default value:</b>	ease
<b>Inherited:</b>	no

**Animatable:** no. [Read about](#) *animatable*

**Version:** CSS3

**JavaScript syntax:** `object.style.transitionTimingFunction="linear"`

## CSS Syntax

transition-timing-function: linear | ease | ease-in | ease-out | ease-in-out | step-start | step-end | steps(int,start | end) | cubic-bezier(*n,n,n,n*) | initial | inherit;

## Property Values

Value	Description
ease	Default value. Specifies a transition effect with a slow start, then fast, then end slowly (equivalent to cubic-bezier(0.25,0.1,0.25,1))
linear	Specifies a transition effect with the same speed from start to end (equivalent to cubic-bezier(0,0,1,1))
ease-in	Specifies a transition effect with a slow start (equivalent to cubic-bezier(0.42,0,1,1))
ease-out	Specifies a transition effect with a slow end (equivalent to cubic-bezier(0,0,0.58,1))
ease-in-out	Specifies a transition effect with a slow start and end (equivalent to cubic-bezier(0.42,0,0.58,1))
step-start	Equivalent to steps(1, start)
step-end	Equivalent to steps(1, end)
steps(int,start   end)	Specifies a stepping function, with two parameters. The first parameter specifies the number of intervals in the function. It must be a positive integer (greater than 0). The second parameter, which is optional, is either the value "start" or "end", and specifies the point at which the change of values occur within the interval. If the second parameter is omitted, it is given the value "end"
<a href="#">cubic-bezier(<i>n,n,n,n</i>)</a>	Define your own values in the cubic-bezier function. Possible values are numeric values from 0 to 1
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

**Tip:** Try the different values in the examples below to understand how it works!



# CSS user-select Property

## Example

Prevent text selection of a <div> element:

```
div {  
  -webkit-user-select: none; /* Safari 3.1+ */  
  -moz-user-select: none; /* Firefox 2+ */  
  -ms-user-select: none; /* IE 10+ */  
  user-select: none; /* Standard syntax */  
}
```

## Definition and Usage

The user-select property specifies whether the text of an element can be selected.

If you double-click on some text, it will be selected/highlighted. This property can be used to prevent this.

<b>Default value:</b>	auto
<b>Inherited:</b>	no
<b>Animatable:</b>	no. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS3
<b>JavaScript syntax:</b>	<i>object.style.userSelect="none"</i>

## CSS Syntax

user-select: auto | none | text | all;

## Property Values

Value	Description	Play it
auto	Default. Text can be selected if the browser allows it	
none	Prevent text selection	
text	The text can be selected by the user	
all	Text selection is made with one click instead of a double-click	

# CSS vertical-align Property

## Example

Vertical align an image:

```
img {  
  vertical-align: text-top;  
}
```

## Definition and Usage

The vertical-align property sets the vertical alignment of an element.

**Default value:** baseline

**Inherited:** no

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS1

**JavaScript syntax:** *object.style.verticalAlign="top"*

## CSS Syntax

vertical-align: baseline | *length* | sub | super | top | text-top | middle | bottom | text-bottom | initial | inherit;

## Property Values

Value	Description	Play it
baseline	Align the baseline of the element with the baseline of the parent element. This is default	
<i>length</i>	Raises or lower an element by the specified length. Negative values are allowed	
%	Raises or lower an element in a percent of the "line-height" property. Negative values are allowed	
sub	Aligns the element as if it was subscript	
super	Aligns the element as if it was superscript	
top	The top of the element is aligned with the top of the tallest element on the line	
text-top	The top of the element is aligned with the top of the parent element's font	
middle	The element is placed in the middle of the parent element	
bottom	The bottom of the element is aligned with the lowest element on the line	

text-bottom	The bottom of the element is aligned with the bottom of the parent element's font
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

## CSS visibility Property

### Example

Make <h2> elements invisible:

```
h2 {
  visibility: hidden;
}
```

### Definition and Usage

The visibility property specifies whether or not an element is visible.

**Tip:** Even invisible elements take up space on the page. Use the [display](#) property to create invisible elements that do not take up space!

**Default value:** visible

**Inherited:** yes

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS2

**JavaScript syntax:** `object.style.visibility="hidden"`

### CSS Syntax

visibility: visible|hidden|collapse|initial|inherit;

### Property Values

Value	Description	Play it
visible	Default value. The element is visible	
hidden	The element is invisible (but still takes up space)	
collapse	Only for table elements. collapse removes a row or column, but it does not affect the table layout. The space taken up by the row or column will be available for other content.	

initial

If collapse is used on other elements, it renders as "hidden"  
Sets this property to its default value. [Read about](#) *initial*

# CSS white-space Property

## Example

Specify that the text in <p> elements will never wrap:

```
p {  
  white-space: nowrap;  
}
```

## Definition and Usage

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

**Default value:** normal

**Inherited:** yes

**Animatable:** no. [Read about](#) *animatable*

**Version:** CSS1

**JavaScript syntax:** *object.style.whiteSpace="nowrap"*

## CSS Syntax

white-space: normal | nowrap | pre | pre-line | pre-wrap | initial | inherit;

## Property Values

Value	Description	Play it
normal	Sequences of whitespace will collapse into a single whitespace. Text will wrap when necessary. This is default	
nowrap	Sequences of whitespace will collapse into a single whitespace. Text will never wrap to the next line. The text continues on the same line until a   tag is encountered	
pre	Whitespace is preserved by the browser. Text will only wrap on line breaks. Acts like the <pre> tag in HTML	
pre-line	Sequences of whitespace will collapse into a single whitespace. Text will wrap when necessary, and on line breaks	

pre-wrap	Whitespace is preserved by the browser. Text will wrap when necessary, and on line breaks
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS word-break Property

## Example

Break words between any two letters:

```
p.test {
  word-break: break-all;
}
```

## Definition and Usage

The word-break property specifies line breaking rules for non-CJK scripts.

**Tip:** CJK scripts are Chinese, Japanese and Korean ("CJK") scripts.

<b>Default value:</b>	normal
<b>Inherited:</b>	yes
<b>Animatable:</b>	no. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS3
<b>JavaScript syntax:</b>	<i>object.style.wordBreak="break-all"</i>

## CSS Syntax

word-break: normal|break-all|keep-all|initial|inherit;

## Property Values

Value	Description
normal	Default value. Break words according to their usual rules
break-all	Lines may break between any two letters
keep-all	Breaks are prohibited between pairs of letters
initial	Sets this property to its default value. <a href="#">Read about initial</a>

inherit      Inherits this property from its parent element. [Read about \*inherit\*](#)

## CSS word-spacing Property

### Example

Specify that the space between words in <p> elements should be 30 pixels:

```
p {  
  word-spacing: 30px;  
}
```

### Definition and Usage

The word-spacing property increases or decreases the white space between words.

**Note:** Negative values are allowed.

**Default value:** normal

**Inherited:** yes

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS1

**JavaScript syntax:** `object.style.wordSpacing="20px"`

### CSS Syntax

word-spacing: normal | *length* | initial | inherit;

### Property Values

Value	Description	Play it
normal	Defines normal space between words (0.25em) . This is default	
<i>length</i>	Defines an additional space between words (in px, pt, cm, em, etc). Negative values are allowed	
initial	Sets this property to its default value. <a href="#">Read about <i>initial</i></a>	
inherit	Inherits this property from its parent element. <a href="#">Read about <i>inherit</i></a>	

## CSS word-wrap Property

## Example

Allow long words to be able to break and wrap onto the next line:

```
p.test {  
  word-wrap: break-word;  
}
```

## Definition and Usage

The word-wrap property allows long words to be able to be broken and wrap onto the next line.

**Default value:** normal

**Inherited:** yes

**Animatable:** no. [Read about animatable](#)

**Version:** CSS3

**JavaScript syntax:** `object.style.wordWrap="break-word"`

## CSS Syntax

word-wrap: normal|break-word|initial|inherit;

## Property Values

Value	Description
normal	Break words only at allowed break points
break-word	Allows unbreakable words to be broken
initial	Sets this property to its default value. <a href="#">Read about initial</a>
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>

# CSS z-index Property

## Example

Set the z-index for an image:

```
img {  
  position: absolute;  
  left: 0px;  
  top: 0px;
```

```
z-index: -1;  
}
```

## Definition and Usage

The z-index property specifies the stack order of an element.

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

**Note:** z-index only works on positioned elements (position:absolute, position:relative, or position:fixed).

**Default value:** auto

**Inherited:** no

**Animatable:** yes. [Read about animatable](#)

**Version:** CSS2

**JavaScript syntax:** `object.style.zIndex="-1"`

## CSS Syntax

z-index: auto|*number*|initial|inherit;

## Property Values

Value	Description	Play it
auto	Sets the stack order equal to its parents. This is default	
<i>number</i>	Sets the stack order of the element. Negative numbers are allowed	
initial	Sets this property to its default value. <a href="#">Read about initial</a>	
inherit	Inherits this property from its parent element. <a href="#">Read about inherit</a>	