

CSS Cheat Sheet (A - Z)

A

- **align-content** - property modifies the behavior of the flex-wrap property.

Values – stretch | center | flex-start | initial | flex-end | space-between | space-around | inherit; [Know here](#)

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

Values – stretch | center | flex-start | flex-end | baseline | initial | inherit; [Know here](#)

- **align-self** - specifies the alignment for the 'selected' item inside the flexible container.

Values – auto | stretch | center | flex-start | flex-end | baseline | initial | inherit; [Know here](#)

- **all** - resets all properties to their initial or inherited value.

Values – initial | inherit | unset;

- **animation:**

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

Values:

- 1) animation-name
- 2) animation-duration
- 3) animation-timing-function
- 4) animation-delay
- 5) animation-iteration-count
- 6) animation-direction
- 7) animation-fill-mode
- 8) animation-play-state

- **animation-name** - specifies a name for the @keyframes animation.

Values – keyframename

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

Values – time; [Know here](#)

- **animation-timing-function** - specifies the speed curve of an animation.

Values – 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; [Know here](#)

- **animation-delay** - specifies a delay for the start of an animation.

Values - time (can be negative like -2s); [Know here](#)

- **animation-iteration-count** - specifies the number of times an animation should be played.

Values - number | infinite; [Know here](#)

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

Values – normal | reverse | alternate | alternate-reverse; [Know here](#)

- **animation-fill-mode** property specifies a style for the element when the animation is not playing (before it starts, after it ends, or both). [Know here](#)

Values – none | forwards | backwards | both;

CSS Cheat Sheet (A - Z)

- **animation-play-state** - specifies whether the animation is running or paused.

Values – paused | running; [Know here](#)



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

Values – visible | hidden; [Know here](#)

- **background:**

Syntax - *bg-color | bg-image | position | bg-size | bg-repeat | bg-origin | bg-clip | bg-attachment;*

Values –

1. background-color
2. background-image
3. background-position
4. background-size
5. background-repeat
6. background-origin
7. background-clip
8. background-attachment

- **background-color** - sets the background color of an element.

Values – color | transparent; [Play it!](#)

- **background-image** - sets one or more background images for an element.

Values – url("xyz"); [Know here](#)

- **background-position** - sets the starting position of a background image.

Values – left top | right center | right bottom | center top etc. | x% y% | xpx ypx; [Know here](#)

- **background-size** - specifies the size of the background images.

Values – auto | length | cover | contain; [Know here](#)

- **background-repeat** - sets if/how a background image will be repeated.

Values – repeat | repeat-x | repeat-y | no-repeat | space | round; [Know here](#)

- **background-origin** property specifies the origin position of a background image

Values – padding-box | border-box | content-box; [Know here](#)

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

Values – border-box | padding-box | content-box; [Know here](#)

- **background-attachment** - sets whether a background image scrolls with the rest of the page, or is fixed. [Know here](#)

Values – scroll | fixed | local;

- **background-blend-mode** - defines the blending mode of each background layer.

Values – normal | multiply | screen | overlay | darken | lighten | color-dodge | saturation | color | luminosity; [Know here](#)

- **border:**

Syntax - *border-width | border-style | border-color;*

Values –

1. border-color: [Know here](#)
2. border-width: [Know here](#)
3. border-style (required): [Know here](#)

CSS Cheat Sheet (A - Z)

- **border-bottom** - is a shorthand property for:

1. **border-bottom-width** [Know here](#)
2. **border-bottom-style** [Know here](#)
3. **border-bottom-color** [Know here](#)
(same for border top/right/left)

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

Values – separate | collapse | initial | inherit; [Know here](#)

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

1. **border-image-source**
2. **border-image-slice** [Know here](#)
3. **border-image-width** [Know here](#)
4. **border-image-outset** [Know here](#)
5. **border-image-repeat** [Know here](#)

- **border-radius** - defines the radius of the element's corners.

Values – [Know here](#)

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

Values – length; [Know here](#)

- **bottom** - affects the vertical position of a positioned element.

Values – auto | length; [Know here](#)

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

Values – none | h-offset v-offset blur spread color | inset; [Know here](#)

- **box-sizing** - defines how the width and height of an element are calculated: should they include padding and borders, or not.

Values – content-box | border-box; [Know here](#)



- **caret-color** - specifies the color of the cursor (caret) in inputs, textareas, or any element that is editable.

Values – auto | color; [Know here](#)

- **clear** - specifies on which sides of an element floating elements are not allowed to float.

Values – none | left | right | both; [Know here](#)

- **clip** - lets you specify a rectangle to clip an absolutely positioned element.

Values – auto | shape; [Know here](#)

- **clip-path** - lets you clip an element to a basic shape or to an SVG source.

Values – clip-source | basic-shape | margin-box | border-box | padding-box | content-box | fill-box | stroke-box | view-box | none; [Know here](#)

- **color** - specifies the color of text.

Values – color; [Know here](#)

columns - is a shorthand property for:

Syntax – auto | column-width column-count;

1. **column-width**: [Know here](#)
2. **column-count**: [Know here](#)

CSS Cheat Sheet (A - Z)

- **column-gap** - specifies the gap between the columns.

Values – length | normal; [Know here](#)

- **column-rule** - sets the width, style, and color of the rule between columns.

Syntax - column-rule-width column-rule-style column-rule-color;

1. column-rule-width: [Know here](#)
2. column-rule-style (required): [Know here](#)
3. column-rule-color: [Know here](#)

- **column-span** - specifies how many columns an element should span across.

Values – none | all; [Know here](#)

- **content** - is used with the ::before and ::after pseudo-element, to insert generated content. [Know here](#)

- **cursor** - specifies the mouse cursor to be displayed when pointing over an element.

[See example](#)

D

- **direction** - specifies the text direction/writing direction within a block-level element.

Values – ltr | rtl; [Know here](#)

- **display** - specifies the display behavior of an element.

[See example](#)

F

- **filter** - defines visual effects (like blur and saturation) to an element (often).

Values – none | blur() | brightness() | contrast() | drop-shadow() | grayscale() | hue-rotate() | invert() | opacity() | saturate() | sepia() | url(); [Know here](#)

- **flex** - sets the flexible length on flexible items.

Syntax - flex-grow flex-shrink flex-basis | auto;

- flex-grow: [Know here](#)
- flex-shrink: [Know here](#)
- flex-basis: [Know here](#)

- **flex-flow** - is a shorthand property for:

1. flex-direction: [Know here](#)
2. flex-wrap: [Know here](#)

Syntax – flex-direction flex-wrap;

- **float** - specifies how an element should float.

Values – none | left | right; [Know here](#)

- **font** - is a shorthand property for:

1. font-style: [Know here](#)
2. font-variant: [Know here](#)
3. font-weight: [Know here](#)
4. font-size: [Know here](#)
5. font-family: [Know here](#)

Syntax – font-style font-variant font-weight font-size font-family|caption|icon|menu|message-box|small-caption|status-bar;

CSS Cheat Sheet (A - Z)



- **gap** - defines the size of the gap between the rows and columns.

Syntax – row-gap column-gap;

- 1) row-gap: [Know here](#)
- 2) column-gap: [Know here](#)

-
- **grid** - is a shorthand property for: [Know here](#)

- 1) grid-template-rows: [Know here](#)
- 2) grid-template-columns: [Know here](#)
- 3) grid-template-areas: [Know here](#)
- 4) grid-auto-rows: [Know here](#)
- 5) grid-auto-columns: [Know here](#)
- 6) grid-auto-flow: [Know here](#)

Syntax – none | grid-template-rows / grid-template-columns | grid-template-areas | grid-template-rows / [grid-auto-flow] grid-auto-columns | [grid-auto-flow] grid-auto-rows / grid-template-columns

-
- **grid-area** - specifies a grid item's size and location in a grid layout.

- 1) grid-row-start: [Know here](#)
- 2) grid-column-start: [Know here](#)
- 3) grid-row-end: [Know here](#)
- 4) grid-column-end: [Know here](#)

Syntax – grid-row-start / grid-column-start / grid-row-end / grid-column-end | itemname;



- **height** – sets the height of an element. (same for width)

Values – auto | length; [Know here](#)



- **isolation** - defines whether an element must create a new stacking content.

Values – auto | isolate; [Know here](#)



- **justify-content** - aligns the flexible container's items when the items do not use all available space on the main-axis (horizontally).

Values – flex-start | flex-end | center | space-between | space-around | space-evenly; [Know here](#)



- **@keyframes** - rule specifies the animation code. [Know here](#)



- **left** - affects the horizontal position of a positioned element. This property has no effect on non-positioned elements. (same for right and top)

Values – auto | length; [Know here](#)

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

CSS Cheat Sheet (A - Z)

Values – normal | length; [Know here](#)

- **line-height** - specifies the height of a line.

Values – normal | number | length;
[Know here](#)

- **list-style** - is a shorthand for the following properties:

1. list-style-type: [Know here](#)
2. list-style-position: [Know here](#)
3. list-style-image: [Know here](#)

Syntax – list-style-type list-style-position list-style-image;



- **margin** - sets the margins for an element. [Know here](#)

1. margin-top: [Know here](#)
2. margin-right: [Know here](#)
3. margin-bottom: [Know here](#)
4. margin-left: [Know here](#)

Syntax – length | auto;

- **max-height** - defines the maximum height of an element.

Values – none | length; [Know here](#)

(same for max-width, min-height, min-width)

- **@media** - @media rule is used in media queries to apply different styles for different media types/devices.

[See example](#)



- **object-fit** - is used to specify how an or <video> should be resized to fit its container.

Values – fill | contain | cover | scale-down | none;

- **object-position** - is used together with object-fit to specify how an or <video> should be positioned with x/y coordinates inside its "own content box".

Values – position; [Know here](#)

- **opacity** - sets the opacity level for an element.

Values – number(from 0 to 1);

- **order** - specifies the order of a flexible item relative to the rest of the flexible items inside the same container.

[See example](#)

- **outline** - is a shorthand property for:

1. outline-width: [Know here](#)
2. outline-style (required): [Know here](#)
3. outline-color: [Know here](#)

Syntax – outline-width outline-style outline-color;

CSS Cheat Sheet (A - Z)

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

[See example](#)

- **overflow** - specifies what should happen if content overflows an element's box.

Values – visible | hidden | scroll | auto; [Know more](#)

(same for overflow-x and overflow-y)

P

- **padding** - it creates extra space within an element, while margin creates extra space around an element. [Know more](#)

- 1) **padding-top**: [Know more](#)
- 2) **padding-right**: [Know more](#)
- 3) **padding-bottom**: [Know more](#)
- 4) **padding-left**: [Know more](#)

- **perspective** - is used to give a 3D-positioned element some perspective.

[See example](#)

- **perspective-origin** - defines at from which position the user is looking at the 3D-positioned element.

[See example](#)

- **position** - specifies the type of positioning method used for an element. [Know here](#)

Values – static | absolute | fixed | relative | sticky; [Know more](#)

R

- **resize** - defines if (and how) an element is resizable by the user.

Values – none | both | horizontal | vertical; [Know more](#)

S

- **scroll-behavior** - specifies whether to smoothly animate the scroll position, instead of a straight jump, when the user clicks on a link within a scrollable box.

Values – auto | smooth; [Know more](#)

T

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

Values – left | right | center | justify; [Know more](#)

- **text-decoration** - specifies the decoration added to text.

Syntax – text-decoration-line text-decoration-color text-decoration-style;

1. **text-decoration-line** (required): [Know more](#)
2. **text-decoration-color**: [Know more](#)
3. **text-decoration-style**: [Know more](#)

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

[See example](#)

CSS Cheat Sheet (A - Z)

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

[See example](#)

- **text-shadow** - adds shadow to text.

Syntax – *h-shadow v-shadow blur-radius color* | none; [Know more](#)

- **text-transform** - controls the capitalization of text.

Values – none | capitalize | uppercase | lowercase; [Know more](#)

- **transform** - applies a 2D or 3D transformation to an element.

Values – none | *transform-functions*;

See all transformations [here](#)

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

Syntax – *x-axis y-axis z-axis*; [Know more](#)

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

Values – flat | preserve-3d;

- **transition** - is a shorthand property for:

1. **transition-property**: [Know more](#)
2. **transition-duration**: [Know more](#)
3. **transition-timing-function**: [Know more](#)
4. **transition-delay**: [Know more](#)

Syntax – *property duration timing-function delay*;

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

[See example](#)



- **vertical-align** - sets the vertical alignment of an element.

Values – baseline | *length* | sub | super | top | text-top | middle | bottom | text-bottom. [Know more](#)

- **visibility** - specifies whether or not an element is visible.

Values – visible | hidden | collapse; [Know more](#)



- **white-space** - specifies how white-space inside an element is handled.

Values – normal | nowrap | pre | pre-line | pre-wrap; [Know more](#)

- **word-break** - specifies how words should break when reaching the end of a line.

Values – normal | break-all | keep-all | break-word; [Know more](#)

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

Values – normal | *length*; [Know more](#)

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

Values – normal | break-word; [Know more](#)



CSS Cheat Sheet (A - Z)

- **writing-mode** - specifies whether lines of text are laid out horizontally or vertically.

Values – horizontal-tb | vertical-r1 | vertical-lr; [Know more](#)



- **z-index** - specifies the stack order of an element.

Values – auto | number; [Know more](#)

ADDITIONAL-

- **background-image**: linear-gradient(red, yellow);
- **background-image**: linear-gradient(to right, red , yellow);
- **background-image**: linear-gradient(180deg, red, yellow);
- Adding variables: [Here](#)

```
:root{--blue:#1e90ff;  
--white:#ffffff;}
```

Use: `var(--blue);`

- Adding Audio: [Here](#)

By - Swapnil Bhagwat

Source - w3schools.com