



PROPERTIES AND VALUES

CSS PROPERTIES

CSS properties are the styles used on specified selectors. They are written before values in the CSS ruleset and are separated from property values by a colon. Different HTML selectors and elements have different properties. Some properties are universal and can be used on every selector. Others work only on specific selectors and under particular conditions.

An example of that is `grid-template-columns`, which is used to style the layout of a page. It works mainly with *divs* with their *display* property set to the grid (`display: grid`).

There are many properties and their values for HTML selectors. According to CSS Tricks, there are "520 distinct property names from 66 technical reports and 66 editor's drafts".

Here are four common properties to work with

- List properties
- Font properties
- Border properties
- Text properties

These properties are common because they are frequently used in all CSS documents and can be applied to different selectors. One unique thing about properties is that they have more than one value attached to them. Text-transform, for example, a property that controls a text's capitalization, can be set to uppercase, lowercase, capitalize, or none. But this also poses a restraint. You must specify a value to the right property else nothing happens. If we have 'text-transform: underline;' nothing will change on the text part because underline is a value for text-decoration.

TEXT PROPERTIES

Properties	Description	Values
color	Sets the color of a text	<i>Hex, RGB, keyword</i>
text-transform	Sets the capitalization of the text	<i>uppercase, lowercase, capitalize, none</i>
text-align	Sets the alignment of the text on the screen	<i>right, left, center, justify</i>
letter-spacing	Sets the spacing between text characters	<i>normal, length</i>
text-decoration	Sets the decoration added to the text	<i>none, underline, line-through, overline</i>

BORDER PROPERTIES

Properties	Description	Values
border	Sets the shorthand combination for border-width, border-style and border-color	<i>border-width, border-style, border-color</i>
border-color	Sets the color for the border	<i>Keyword, RGB, Hex, transparent, inherit</i>
border-radius	Sets the radius of the four corners of an element's border	<i>length, percentage, initial, inherit</i>
border-style	Sets the style for an element's border	<i>none, hidden, dotted, solid, dashed, double, groove, inset, outset, ridge, initial, inherit</i>
border-image	Sets an image as an element's border	<i>border-image-source, border-image-width, border-image-slice, border-image-repeat, border-image-outset, initial, inherit</i>

FONT PROPERTIES

Properties	Description	Values
font	Sets the shorthand for all the font specifications	<i>font-style, font-variant, font-weight, font-size/line-height, font-family, caption, icon, menu, message-box, small-caption, status-bar, inherit</i>
font-weight	Sets the weight of a font	<i>normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900, inherit</i>
font-style	Sets the style of a font	<i>Normal, italic, oblique, initial, inherit</i>

LIST PROPERTIES

Properties	Description	Values
list-style	Shorthand combination for list-style-type, list-style-position, and list-style-image	<i>list-style-type, list-style-position, list-style-image, inherit</i>
list-image	Sets an image as the list-item marker	<i>none, url, initial, inherit</i>
list-type	This sets the type of list-item marker	<i>none, disc, circle, square, decimal, decimal-leading-zero, armenian, georgian, lower-alpha, upper-alpha, lower-greek, lower-latin, upper-latin, lower-roman, upper-roman, inherit</i>

CSS VALUES

Values are written immediately after the colon that separates them from CSS properties. CSS values aren't just text; they come in different forms - URLs, units, measurements, integers, strings, inherit, auto, none, etc. We will look at different CSS values and how to implement them.

TEXT

Text values are prevalent in CSS. They are usually written without quotes, unlike strings. Every valid CSS property has a text as a value. Margin takes units as values, but it also has auto, which in this case, is taken as text.

```
h1 {  
  color: red;  
  text-align: center;  
}
```

Red and center are text values of color and text-align.

INTEGERS

Integers are numbers from one to zero or zero to nine. Some CSS properties make allowance for their values to be defined as integers, even negative integers. The values for properties like column-count and z-index are defined using integers.

```
div {  
  column-count: 4;  
}  
div {  
  z-index: -1;  
}
```

UNITS/MEASUREMENTS

Because CSS must be used to position items on a web page, general layout, and media queries, many properties take units and measurements as their value. There are many property-specific units for CSS values, but available units like px, em, fr and percentages are standard.

```
h3 {  
  font-size: 24px;  
}  
div{  
  grid-template-columns: 1fr 1fr 1fr 1fr;  
}  
div {  
  width: 100%;  
  font-size: 2em;  
}
```

URLS

Properties like background-image take an actual URL as their value. This URL could be absolute or relative. Meaning, you could copy the address of an image online and use it as your background image, or you could get images from the project directory/folder you are working with on your computer.

```
div{  
  background-image: url("https://cdn.pixabay.com/photo/2015/04/23/22/00/tree-736885_340.jpg");  
}  
div{  
  background-image: url("../Images/landscape.jpg");  
}
```

STRINGS

Strings are text within quotes. The content property which you can use to insert generated content allows strings within its parameter.

```
.div :: after {  
  content: "This is me";  
}
```

COLORS

The color/background-colors are properties used to set the color of HTML elements. Their values accept either a hexadecimal color combination value, an RGB specification, or a defined color.

```
h2 {
  color: red;
}
div{
  background-color: #000000;
}
```

AUTO, NONE, INHERIT

Auto, *None*, and *Inherit* are keywords values in CSS. Auto value allows the property to fill or adjust according to the content of the element.

```
div {
  width: auto;
  margin: auto;
}
```

None adds no value to the specified property.

```
div{
  border: none;
}
```

Inherit keyword allows the property to inherit the same value from the parent element. All CSS properties accept *inherit*.

```
<div>
  <p> this is me </p>
  <p id="pickme"> this is not me </p>
</div>
p{
  color: red;
}
#pickme p {
  color: inherit;
}
```

Although all the examples above are of properties with one value, CSS values can have more than one value, and you can manipulate it to write shorthand. Shorthand in CSS is a shorter, more compact way of setting values for similar but multiple properties.

These values,

```
div{
  border-style: solid;
  border-color: red;
  border-width: 5px;
}
```

can be rewritten in shorthand as

```
div{
  border: 5px solid red;
}
```

Other properties with multiple values include, but not limited to

```
margin: 1px 3px 6px 2px; /* defining the top, left, bottom and left of a margin */
font: italic small-caps bold 12px Georgia, serif; /*defining font-style, font-
variant-caps, font-weight, font-size, and font-family. */
box-shadow: 5px 2px 8px #aaaaaa; /* defining the horizontal offset, vertical offset,
blur, and color values */
```