

Advanced CSS Training

CSS Shorthand Properties

Lesson 1, Activity 2: What is CSS Shorthand?

A CSS shorthand property is a property that can take multiple values, with each value relating to a different regular CSS property. Values are separated by spaces as shown below:

Syntax

```
selector {
  property: value1 value2 value3;
}
```

In this lesson, we will cover the following shorthand properties:

- margin
- padding
- border
 - border-left
 - border-right
 - border-top
 - border-bottom
 - border-color
 - border-style
 - border-width
- list-style
- background
- font

Benefits of Using CSS Shorthand

There are two major benefits of using CSS shorthand properties:

- Less code for you to write.
- Smaller files for the user to download.

Value Order of CSS Shorthand Properties

Some properties, such as `border`, allow you to rearrange values without affecting the display of the style. For example:

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

...will display the same as...

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

However, for other properties, such as `padding` and `margin`, order does matter as we'll see later in the lesson.

CSS Initial Values

Many CSS properties have *initial values*, which are specified in the [CSS 2.0 specification](#). Knowing how CSS initial values affect your styles will help you make your CSS shorthand code even shorter by omitting them from a declaration.

Omitting values from a shorthand property declaration will result in one of the following:

- CSS inserts a default called an initial value.
- The omitted value may be inherited from elsewhere in your CSS.

As an example, the initial values for the `border` property are shown below:

CSS Initial Values for the `border` Property

Property	Initial Value
<code>border-width</code>	medium
<code>border-style</code>	none
<code>border-color</code>	inherited from a <code>color</code> property

Note that the initial value for the `border` style is "none". No border will appear unless you change this value even if you assign a `border-width` and `border-color`.

The following example demonstrates how you can remove values from the `border` property relying on CSS initial values as well as styles inherited from other property values.

Code Sample:

[CssShorthand/Demos/OmittingPropertyValues.html](#)

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="UTF-8">
<title>Omitting CSS Property Values</title>
<style type="text/css" media="screen">
body {
  color: red;
}

p.one {
  border: #000 solid 1px; /* sets a black 1px width, solid style border. */
```

```

    color: blue;
}

p.two {
    border: solid 1px; /* border color value omitted so it inherits from the color property below (blue). */
    color: blue;
}

p.three {
    border: solid 1px; /* border color inherited from the body */
}
</style>
</head>

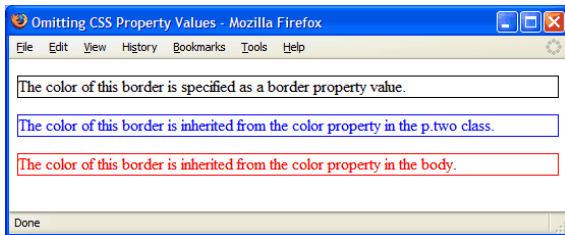
<body>
<p class="one">The color of this border is specified as a border property value.</p>

<p class="two">The color of this border is inherited from the color property in the p.two class.</p>

<p class="three">The color of this border is inherited from the color property in the body.</p>
</body>
</html>

```

The above code will render the following:



We will indicate the initial values of properties as we introduce them throughout the lesson.

Lesson 1, Activity 3: Shorthand Margin and Padding

The margin and padding shorthand properties interpret values in the same way:

- One value: sets all four sides uniformly.
- Two values: first value sets top and bottom, second value sets right and left.
- Three values: first value sets top, second value sets both right and left, third value sets bottom.
- Four values: sets the top, right, bottom and left, respectively.

Syntax

```
selector {
  padding: AllSides;
}
selector {
  padding: TopAndBottom RightAndLeft;
}
selector {
  padding: Top RightAndLeft Bottom;
}
selector {
  padding: Top Right Bottom Left;
}
```

The initial value for each margin and padding property is 0.

The following example demonstrates how you can use the shorthand properties for padding.

Code Sample:

[CssShorthand/Demos/Padding.html](#)

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="UTF-8">
<title>CSS Margin and Padding Shorthand</title>
<style type="text/css" media="screen">
div {
  border: 1px solid black;
  margin: .75em;
}

div.one {
  padding: 1em; /* sets the top, right, bottom and left padding to 1em. */
}

div.two {
  padding: 1em 2em; /* sets the top and bottom to 1em and the right and left padding to 2em. */
}

div.three {
  padding: 1em 2em 3em; /* sets the top to 1em, right and left to 2em, and bottom to 3em. */
}

div.four {
  padding: 1em 2em 3em 4em; /* sets the top to 1em, right to 2em, bottom to 3em, and left to 4em. */
}
</style>
</head>

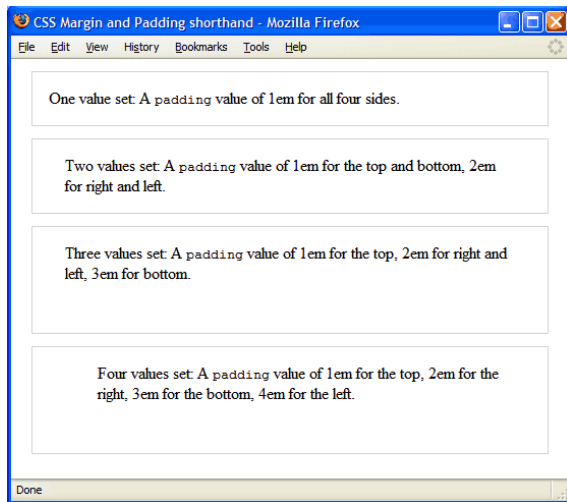
<body>
<div class="one">One value set: A <code>padding</code> value of 1em for all four sides.</div>

<div class="two">Two values set: A <code>padding</code> value of 1em for the top and bottom, 2em for right and left.</div>

<div class="three">Three values set: A <code>padding</code> value of 1em for the top, 2em for right and left, 3em for bottom.</div>

<div class="four">Four values set: A <code>padding</code> value of 1em for the top, 2em for the right, 3em for the bottom, 4em for the left.</div>
</body>
</html>
```

The above code will render the following:



Lesson 1, Activity 4: Shorthand Border

The shorthand for `border` is shown below:

Syntax

```
selector {
  border: border-width border-style border-color;
}
```

The shorthand for `border` can also be applied individually to `border-top`, `border-right`, `border-bottom` and `border-left`. For example, the following syntax demonstrates the shorthand structure applied to a `border-top` property.

Syntax

```
selector {
  border-top: border-width border-style border-color;
}
```

CSS Initial Values for the `border` Property

Property	Initial Value
<code>border-width</code>	medium
<code>border-style</code>	none
<code>border-color</code>	inherited from a <code>color</code> property

The following example demonstrates the use of the `border` shorthand property.

Code Sample:

[CssShorthand/Demos/Border.html](#)

```
---- C O D E   O M I T T E D ----
div.one {
  border: 1px solid blue; /* This sets the border value to a solid blue border with a width of 1px. */
}

div.two {
  border-top: 1px solid blue; /* This sets a solid blue border with a width of 1px on the top . */
  border-right: 2px solid blue; /* This sets a solid blue border with a width of 2px on the right. */
  border-bottom: 3px solid blue; /* This sets a solid blue border with a width of 3px on the bottom. */
  border-left: 4px solid blue; /* This sets a solid blue border with a width of 4px on the left. */
}

div.three {
  border: solid; /* This border uses the initial values for color and width. */
}

div.four {
  border: 1px #00f; /* No style is set for this border so it will not display. */
}
</style>
</head>

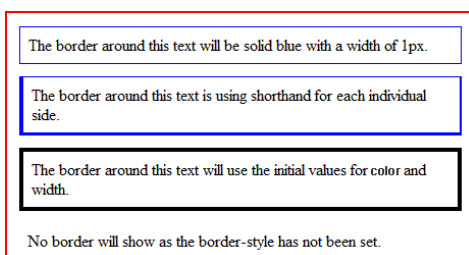
<body>
  <div class="one">The border around this text will be solid blue with a width of 1px.</div>

  <div class="two">The border around this text is using shorthand for each individual side.</div>

  <div class="three">The border around this text will use the initial values for color and width.</div>

  <div class="four">No border will show as the border-style has not been set.</div>
</body>
</html>
```

The above code will render the following:



Other Border Shorthand Properties

In the list of available shorthand properties, you may have noticed `border-color`, `border-style`, `border-width`. These shorthand properties shouldn't be confused with the `border` property because their structure is different. The structure for these properties is similar to `margin` and `padding` in that you set the value for each side.

The code below shows the syntax for `border-color`, `border-style`, and `border-width`:

Syntax

```
selector {  
  border-color: Top Right Bottom Left;  
  border-style: Top Right Bottom Left;  
  border-width: Top Right Bottom Left;  
}
```

Lesson 1, Activity 5: Shorthand Backgrounds

The background property is used to set the individual properties for background-color, background-image, background-repeat, background-attachment and background-position. Note that order matters (although some browsers may show some leniency).

Syntax

```
selector {
  background: background-color background-image background-repeat background-attachment background-position;
}
```

CSS Initial Values for the background Property

Property	Initial Value
background-color	transparent
background-image	none
background-repeat	repeat
background-attachment	scroll
background-position	0% 0%

When using shorthand for the background property, there are some things to be aware of:

- The background color renders first then the background image is placed over the color. When we cover CSS Background Tricks, we'll see that this can be exploited.
- You can use background in place of background-color to set the background color (e.g. background: red;).
- You can use background in place of background-image to set the background image (e.g. background: url(bg.gif);).

The following example demonstrates the background shorthand property.

Code Sample:

[CssShorthand/Demos/Background.html](#)

```
---- C O D E   O M I T T E D ----
div.one {
  background: #ccc; /* shorthand for background-color */
}

div.two {
  background: url('Images/block.gif'); /* shorthand for background-image */
  color: white;
}

div.three {
  background: #ccc url('Images/block.gif') repeat-x scroll 0% 0%; /* uses all available values */
  color: white;
}

div.four {
  background: #ccc url('Images/block.gif') no-repeat scroll 20% 75%; /* different background position values */
}

div.five {
  background: #ccc url('Images/block.gif') no-repeat scroll bottom center; /* and still different background position values */
}
</style>
</head>

<body>
<div class="one">This background sets only the color.</div>

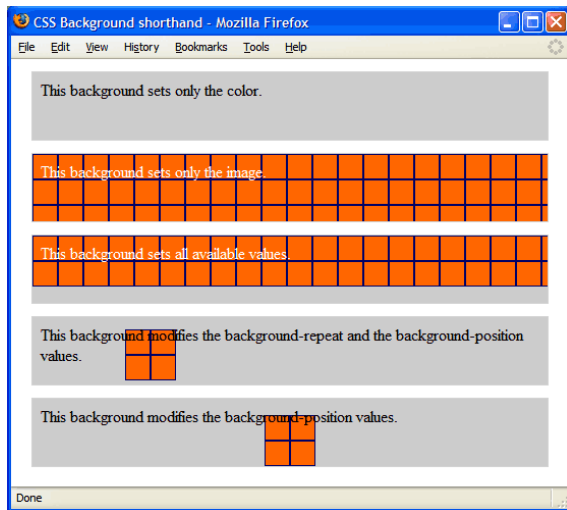
<div class="two">This background sets only the image.</div>

<div class="three">This background sets all available values.</div>

<div class="four">This background modifies the background-repeat and the background-position values.</div>

<div class="five">This background modifies the background-position values.</div>
</body>
</html>
```

The above code will render the following:



Lesson 1, Activity 6: Shorthand Fonts

The `font` property, which is well supported, is used to set the individual properties for `font-style`, `font-variant`, `font-weight`, `font-size`, `line-height` and `font-family`.

Syntax

```
selector {
  font: font-style font-variant font-weight font-size/line-height font-family;
}
```

When using shorthand for the `font` property, note that:

- Order matters.
- You must set values for `font-size` and `font-family` in that order.
- When setting a font name with spaces in the name, e.g. Times New Roman, you must enclose it within quotation marks.
- Values for `font-size` and `font-family` are required. All other values are optional. If omitted, the initial values will be used.
 - Initial values can trick you here. If you use the `font` property for `h1`, which is bold by default, but you don't assign a value for the `font-weight` portion, the initial value of "normal" will be used and the `h1` will not appear bold.
- When setting the `font-size` and `line-height` in the same declaration, they must be side by side with a slash between the two values as shown below:

Syntax

```
selector {
  font: font-size/line-height;
}
```

CSS Initial Values for the `font` Property

Property	Initial Value
<code>font-style</code>	normal
<code>font-variant</code>	normal
<code>font-weight</code>	normal
<code>font-size</code>	medium
<code>line-height</code>	normal
<code>font-family</code>	browser default

Code Sample:

[CssShorthand/Demos/Font.html](#)

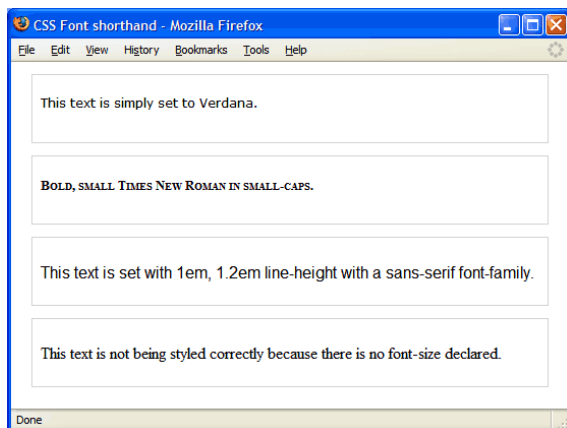
```
---- CODE OMITTED ----
p.one {
  font: .75em verdana; /* At a minimum, you'll need to set font-size and font-family in that order. */
}

p.two {
  font: bold small-caps small/110% "Times New Roman", serif; /* Sets the font-weight, font-variant, font-size/line-height, and font-family with both a font name and serif. */
}

p.three {
  font: 1em/1.2em sans-serif; /* Sets the font-size/line-height and font-family. */
}

p.four {
  font: sans-serif; /* Incorrect styling - font-size required*/
}
</style>
---- CODE OMITTED ----
```

The above code will render the following:



Lesson 1, Activity 7: Shorthand Lists

The `list-style` property is used to set the `list-style-type`, the `list-style-position` and the `list-style-image` properties. The order of values is not important.

Syntax

```
selector {
  list-style: list-style-type list-style-position list-style-image;
}
```

CSS Initial Values for the `list-style` Property

Property	Initial Value
<code>list-style-type</code>	<code>disc/decimal</code>
<code>list-style-position</code>	<code>outside</code>
<code>list-style-image</code>	<code>none</code>

Code Sample:

[CssShorthand/Demos/ListStyle.html](#)

```
---- CODE OMITTED ----
ul.one {
  /* No styling applied. */
}

ul.two {
  list-style: none; /* This sets the list-style-type to none removing the disc. */
}

ul.three {
  list-style: url('Images/listArrow.gif'); /* This list uses an image. */
}

ul.four {
  list-style: square url('Images/ImageNotReal.gif'); /* This list uses an image. If the image file is not found, the list will resort to square as the list-style-type value. */
}

</style>
</head>

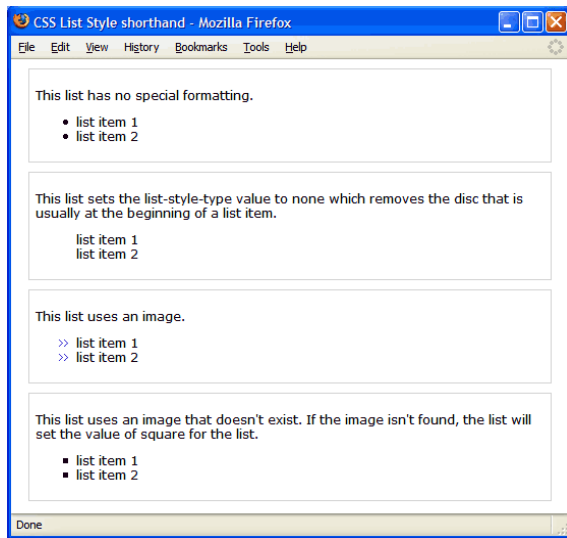
<body>
<div>
  <p>This list has no special formatting.</p>
  <ul class="one">
    <li>list item 1</li>
    <li>list item 2</li>
  </ul>
</div>

<div>
  <p>This list sets the <code>list-style-type</code> value to "none" which removes the disc that is usually at the beginning of a list item.</p>
  <ul class="two">
    <li>list item 1</li>
    <li>list item 2</li>
  </ul>
</div>

<div>
  <p>This list uses an image for the glyph.</p>
  <ul class="three">
    <li>list item 1</li>
    <li>list item 2</li>
  </ul>
</div>

<div>
  <p>This list uses an image that doesn't exist. As the image isn't found, the "backup value" of "square" will be used.</p>
  <ul class="four">
    <li>list item 1</li>
    <li>list item 2</li>
  </ul>
</div>
</body>
</html>
```

The above code will render the following:



Lesson 1, Activity 9: Using Shorthand

Duration: 10 to 20 minutes.

In this exercise, you will modify [CssShorthand/Exercises/Shorthand.css](#) to use shorthand properties wherever possible.

1. Check the file size of [CssShorthand/Exercises/Shorthand.css](#). It should be 1.33KB.
2. Open [CssShorthand/Exercises/Shorthand.css](#) for editing.
3. Modify values to use shorthand where possible.
4. Make sure you have no errors by validating the CSS. Instructions are below.
 1. Select all of your CSS code.
 2. Copy the selected text.
 3. Go to the [W3C CSS Validation Service](#).
 4. Choose the **By direct input** tab if it's not already selected.
 5. Paste the text you selected.
 6. Click the **Check** button.
5. Save the file and check the file size again. There should be a reduction in the size of the file based on how many items you modified.
6. Open [CssShorthand/Exercises/CssShorthand.html](#) and your page should render as follows:

**Solution:**[CssShorthand/Solutions/CssShorthand.css](#)

```

h1, h2, h3, h4, h5, h6 {
    font-family: tahoma;
    color: #00F;
}

p {
    font: .75em verdana, sans-serif;
    color: #000;
}

li {
    font: 1em tahoma, sans-serif;
    color: #000;
}

#header {
    height: 75px;
    background: #fff url('Images/CssShorthandHeader.gif') repeat-x;
    margin: 10px auto 0 auto;
    padding: 10px 5px;
    border-bottom: 1px solid #000;
    font-family: Tahoma;
}

#global-nav {
    float: left;
    width: 23%;
    background: #0F0;
    border: thin dotted #000;
    border-top: 0;
}

#global-nav ul {
    list-style: none;
    margin: 10px 0 0 0;
    padding-left: 0;
}

#global-nav li a {
    display: block;
    text-decoration: none;
    padding: 5px 10px;
}

#global-nav li a:hover {
    background: #fff;
}

#content {
    width: 75%;

```

```
float: right;
padding: 10px 0;
}

#footer {
clear: both;
border-top: 1px solid #000000;
font: small tahoma;
}
```