# **CSS Padding**

In this tutorial you will learn how to adjust the padding area of an element using CSS.

## **CSS Padding Properties**

The CSS padding properties allow you to set the spacing between the content of an element and its border (or the edge of the element's box, if it has no defined border).

The padding is affected by the element's [background-color](https://www.tutorialrepublic.com/css-reference/css-background-color-property.php). For instance, if you set the background color for an element it will be visible through the padding area.

## **Define Paddings for Individual Sides**

You can specify the paddings for the individual sides of an element such as top, right, bottom, and left sides using the CSS padding-top, padding-right, padding-bottom, and the padding-left properties, respectively. Let's try out an example to understand how it works:

#### **Example**

[Try this code »](https://www.tutorialrepublic.com/codelab.php?topic=css&file=padding)

h1 { padding-top: 50px; padding-bottom: 100px; } p { padding-left: 75px; padding-right: 75px; }

The padding properties can be specified using the following values:

* *length* - specifies a padding in px, em, rem, pt, cm, etc.
* *%* - specifies a padding in percentage (%) of the width of the containing element.
* inherit - specifies that the padding should be inherited from the parent element.

Unlike [CSS margin](https://www.tutorialrepublic.com/css-tutorial/css-margin.php), values for the padding properties cannot be negative.

## **The Padding Shorthand Property**

The padding property is a shorthand property to avoid setting padding of each side separately, i.e., padding-top, padding-right, padding-bottom and padding-left.

Let's take a look at the following example to understand how it basically works:

#### **Example**

[Try this code »](https://www.tutorialrepublic.com/codelab.php?topic=css&file=padding-shorthand-property)

h1 { padding: 50px; /\* apply to all four sides \*/ } p { padding: 25px 75px; /\* vertical | horizontal \*/ } div { padding: 25px 50px 75px; /\* top | horizontal | bottom \*/ } pre { padding: 25px 50px 75px 100px; /\* top | right | bottom | left \*/ }

This shorthand notation can take one, two, three, or four whitespace separated values.

* If *one value* is specified, it is applied to **all four sides**.
* If *two values* are specified, the first value is applied to the **top and bottom** side, and the second value is applied to the **right and left** side of the element's box.
* If *three values* are specified, the first value is applied to the **top**, second value is applied to **right and left** side, and the last value is applied to the **bottom**.
* If *four values* are specified, they are applied to the **top**, **right**, **bottom** and the **left** side of the element's box respectively in the specified order.

It is recommended to use the shorthand properties, it will help you to save some time by avoiding the extra typing and make your CSS code easier to follow and maintain.

## **Effect of Padding and Border on Layout**

When creating web page layouts, adding a padding or border to the elements sometimes produce unexpected result, because padding and border is added to the width and height of the box generated by the element, as you have learnt in the [CSS box model chapter](https://www.tutorialrepublic.com/css-tutorial/css-box-model.php).

For instance, if you set the width of a <div> element to 100% and also apply left right padding or border on it, the horizontal scrollbar will appear. Let's see an example:

#### **Example**

[Try this code »](https://www.tutorialrepublic.com/codelab.php?topic=css&file=effect-of-padding-on-element-box-size)

div { width: 100%; padding: 25px; }

To prevent padding and border from changing element's box width and height, you can use the CSS box-sizing property. In the following example the width and height of the <div> box will remain unchanged, however, its content area will decrease with increasing padding or border.