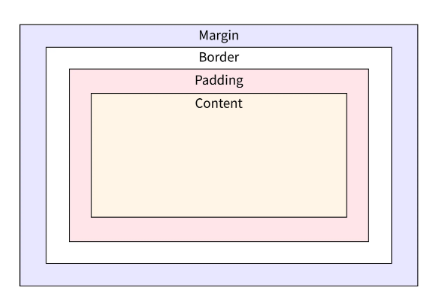
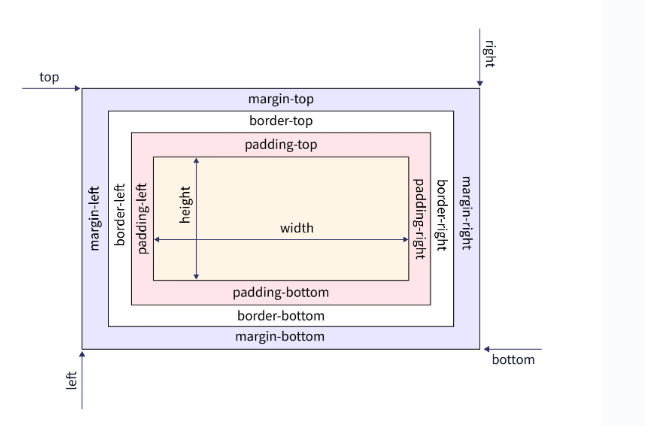
The CSS Box Model

In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content.



* **content**: This property is used to displays the text, images, etc, that can be sized using the [width & height](https://www.geeksforgeeks.org/css-height-and-width/) property.
* [**padding**](https://www.geeksforgeeks.org/css-margins-padding/): This property is used to create space around the element, inside any defined border.
* [**border**](https://www.geeksforgeeks.org/css-borders/): This property is used to cover the content & any padding, & also allows to set the style, color, and width of the border.
* [**margin**](https://www.geeksforgeeks.org/css-margins-padding/): This property is used to create space around the element ie., around the border area.
* **Content** The **content area** contains the real content of the element, such as text, an image, or a video player. It is the area where the content gets displayed on the webpage. Its dimensions can be modified using properties like width and height.
* **Padding** The **padding area** is the space around the content area and within the element's border. It creates extra space inside the element's border and uses the same background as the element itself. The dimensions of the padding is determined by the padding-top, padding-right, padding-bottom, padding-left, and shorthand padding properties.
* **Border** The **border area** is the space around the padding area and within the margin. It includes the element's borders and wraps the content and any padding. Its size and style can be controlled using border and related properties. For example, it can be set to dotted, dashed, solid, double, none, or hidden. It can also have rounded corners using the border-radius property.
* **Margin** The **margin area** is the transparent space outside the element's border and doesn't have any background color. The margin wraps the content, padding, and border and is mostly used to separate the element from other HTML elements on the web page. The size of the margin area is specified using the margin-top, margin-right, margin-bottom, margin-left, and shorthand margin properties.



<!DOCTYPE html>

<head>

<title>CSS Box Model</title>

<style>

.main {

width: 700px;

border: 5px solid black;

}

.box1 {

font-size: 30px;

font-weight: bold;

background-color: chartreuse;

border: 5px solid darkgreen;

margin: 20px;

}

.box2 {

font-size: 50px;

font-weight: bold;

padding: 100px;

// background-color: yellow;

border: 5px solid red;

margin: 100px;

}

</style>

</head>

<body>

<div class="main">

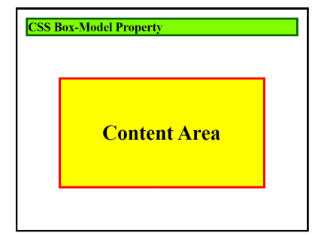
<div class="box1">CSS Box-Model Property</div>

<div class="box2">Content Area</div>

</div>

</body>

</html>



## How To Use the Padding Property in CSS

The padding property may be defined using one, two, three, or four values:

* When one value is specified, it applies the same padding to all four sides.
* When two values are specified, the first padding applies to the top and bottom, the second to the left and right.
* When three values are specified, the first padding applies to the top, the second to the right and left, and the third to the bottom.
* When four values are specified, the paddings apply to the top, right, bottom, and left.

<!DOCTYPE html>

<head>

<title>Padding property</title>

<style>

.main {

width: 400px;

border: 5px solid red;

}

.box{

width: 300px;

height: 50px;

margin: 10px;

font-size: 30px;

font-weight: bold;

}

.box4 {

padding: 30px 15px 20px 25px;

background-color: aqua;

border: 5px solid darkblue;

}

</style>

</head>

<body>

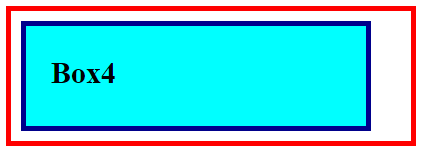
<div class="main">

<div class="box box4">Box4</div>

</div>

</body>

</html>



<!DOCTYPE html>

<head>

<title>Padding property</title>

<style>

.main {

width: 400px;

height: 400px;

border: 5px solid blue;

margin: 30px;

}

.box{

width: 300px;

height: 300px;

margin: 20px;

font-size: 30px;

font-weight: bold;

}

.pd1 {

padding: 20px;

background-color: aqua;

border: 5px solid red;

}

</style>

</head>

<body>

<div class="main">

<div class="box pd1">how are you my friend how

are you my friendhow are you my friendhow

are you my friendhow are you my friendhow are you my friend

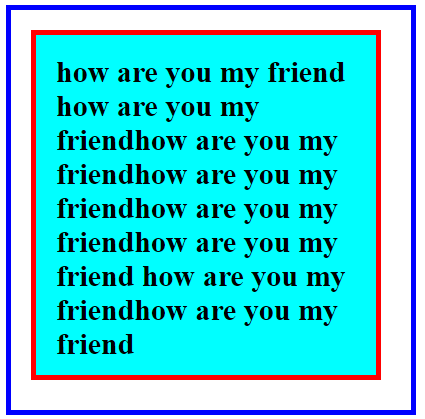
how are you my friendhow are you my friend

</div>

</div>

</body>

</html>



## How to use the Border Property in CSS

The border shorthand CSS property sets an element's border. This border property is a shorthand property for setting *border-color*, *border-style*, and *border-width* in a single declaration. You can use one, two, or three of the values to specify the border property. It doesn't matter what order the values are in.

* **<line-width>** : It sets the width of the border for all four sides of an element. The width can be specified in px, cm, em, etc or by using one of the three keywords: thin, medium, or thick. The default value is medium if not mentioned.
* **<line-style>** : It sets the style of the border for all four sides of an element. The style can be specified using keywords, such as none, dashed, double, solid, etc. The default value is none if not mentioned.
* **<color>** : It sets the color of the border for all four sides of an element. The color can be specified using keywords, such as purple, yellow, etc., or hex and rgb values. The default value is currentcolor if not mentioned.

**Note:** If the border's style isn't specified, it will be invisible. This is because the style is set to none by default.

**<!DOCTYPE html>**

**<head>**

**<title>Border property</title>**

**<style>**

**.box {**

**width: 400px;**

**border: 5px solid black;**

**font-size: 30px;**

**font-weight: bold;**

**}**

**.box1 {**

**margin: 10px;**

**padding: 25px;**

**background-color: chartreuse;**

**border: 5px dashed darkgreen;**

**}**

**.box2 {**

**margin: 10px;**

**padding: 25px;**

**background: gold;**

**border: 5px orangered;**

**}**

**.box3 {**

**margin: 10px;**

**padding: 25px;**

**background-color: violet;**

**border: 10px ridge darkmagenta;**

**}**

**</style>**

**</head>**

**<body>**

**<div class="box">**

**<div class="box1">Box1</div>**

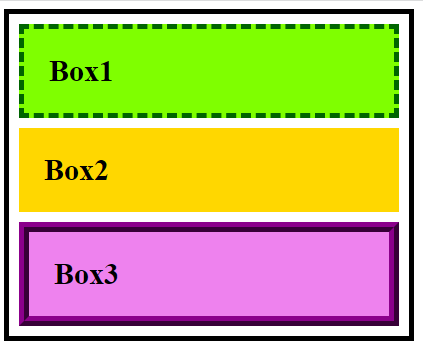
**<div class="box2">Box2</div>**

**<div class="box3">Box3</div>**

**</div>**

**</body>**

**</html>**

****

**Example2**

<html>

<head>

<title>Border property</title>

<style>

.main {

display: grid;

grid-template-rows: 150px 150px;

grid-template-columns: 150px 150px 150px 150px;

grid-gap: 40px;

}

.box {

font-size: 20px;

font-weight: bold;

text-align: center;

padding-top: 45px;

}

.box1 {

border: 10px dotted darkgreen;

}

.box2 {

border: 10px dashed red;

}

.box3 {

border: 10px solid darkblue;

}

.box4 {

border: 10px double chartreuse;

}

.box5 {

border: 10px groove darkmagenta;

}

.box6 {

border: 10px ridge slategrey;

}

.box7 {

border: 10px inset olive;

}

.box8 {

border: 10px outset orange;

}

</style>

</head>

<body>

<div class="main">

<div class="box box1">10px dotted darkgreen</div>

<div class="box box2">10px dashed red</div>

<div class="box box3">10px solid darkblue</div>

<div class="box box4">10px double chartreuse</div>

<div class="box box5">10px groove darkmagenta</div>

<div class="box box6">10px ridge slategrey</div>

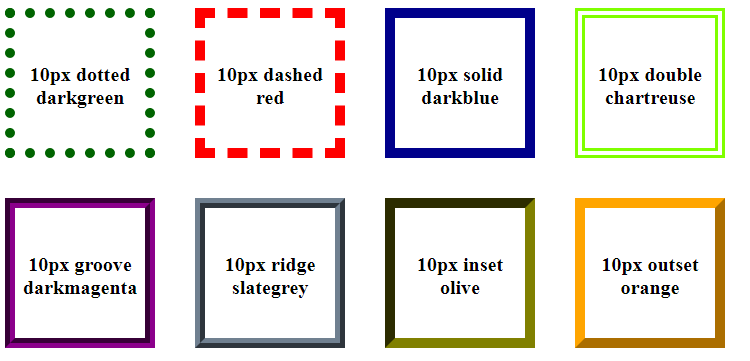
<div class="box box7">10px inset olive</div>

<div class="box box8">10px outset orange</div>

</div>

</body>

</html>



## Inputs of the Border Property

The longhand border properties set the values of the width, style, and color of the border. These are the following border properties of CSS that are used most:

### Border-width

The border-width property of CSS sets the width of an element's border. This property is a shorthand for the following CSS properties:

* border-bottom-width
* border-left-width
* border-right-width
* border-top-width

This border-width property can also be defined using one, two, three, or four inputs, the same as the padding property, and follows the same pattern. The values are either explicit positive values or a keyword. The border-width property can have the following values:

* **length** - specifies the border's width in px, cm, em, rem, etc.
* **keyword** - it can be thin, medium, or thick (Here, thin ≤ medium ≤ thick).

The default value of border-width property is **medium** if not defined by the user.

### Border-style

The border-style property of CSS sets the line style for all four sides of an element's border. This property is a shorthand for the following CSS properties:

* border-top-style
* border-right-style
* border-bottom-style
* border-left-style

This property can also be defined using one, two, three, or four inputs, the same as the padding property, and follows the same pattern. The border-style property can have the following keyword values:

* none
* hidden
* dotted
* dashed
* solid
* double
* groove
* ridge
* inset
* offset

The default value of border-style property is **none** if not defined by the user. It means that if the style of the border is not defined it will be invisible.

### Border-color

The border-color property of CSS sets the color of an element's border. This property is a shorthand for the following CSS properties:

* border-bottom-color
* border-left-color
* border-right-color
* border-top-color

This property can also be defined using one, two, three, or four inputs, the same as the padding property, and follows the same pattern. The border-color property can have the following keyword values:

* **keyword** - specifies the border's color using keywords, such as 'red', 'blue', etc.
* **HEX** - specifies the border's color using hexadecimal values, such as "#800080", #ff0000, etc.
* **RGB** - specifies the border's color using RGB values, such as "rgb(255,0,0)", rgb(123, 255, 0), etc.

The default value of border-color property is **currentColor** if not defined by the user. It means that it inherits the value of color property of the element.

## Calculating Widths and Heights In The Box Model

**The total width taken by the element on the page will be:**

*T*otal element width = width + left padding + right padding + left border + right border + left margin + right margin

**The total height taken by the element on the page will be:**

Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

**<!DOCTYPE html>**

**<head>**

**<title>Calculating Widths and Heights In The Box Model</title>**

**<style>**

**.box {**

**width: 200px;**

**height: 100px;**

**border: 5px solid black;**

**font-size: 20px;**

**font-weight: bold;**

**text-align: center;**

**margin: 30px;**

**padding: 20px;**

**}**

**</style>**

**</head>**

**<body>**

**<div class="box">Box</div>**

**</body>**

**</html>**

the **actual** space taken up by the element on the web page will be 310px wide (200 + 20 + 20 + 5 + 5 + 30 + 30) and 210px high (100 + 20 + 20 + 5 + 5 + 30 + 30).

**Explain the CSS box model along with its different parts. Also, explain, using an example, how you can .calculate the element width and height when you are given the width and height of various parts of the CSS box model.**

# **CSS Box Sizing**

The CSS box-sizing property allows us to include the padding and border in an element's total width and height.

By default, the width and height of an element is calculated like this:

width + padding + border = actual width of an element  
height + padding + border = actual height of an element

This means: When you set the width/height of an element, the element often appears bigger than you have set (because the element's border and padding are added to the element's specified width/height).

The following illustration shows two <div> elements with the same specified width and height:

<!DOCTYPE html>

<html>

<head>

<style>

.div1 {

width: 300px;

height: 100px;

border: 1px solid blue;

}

.div2 {

width: 300px;

height: 100px;

padding: 50px;

border: 1px solid red;

}

</style>

</head>

<body>

<h1>Without box-sizing</h1>

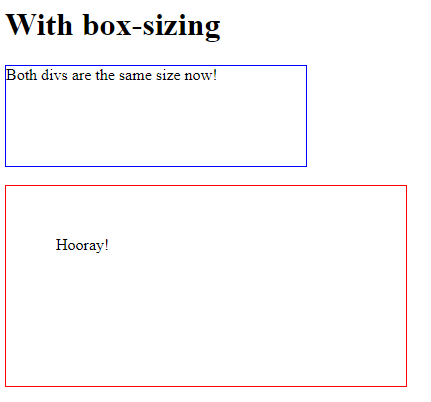
<div class="div1">This div is smaller (width is 300px and height is 100px).</div>

<br>

<div class="div2">This div is bigger (width is also 300px and height is 100px).</div>

</body>

</html>



The box-sizing property solves this problem.

## With the CSS box-sizing Property

The box-sizing property allows us to include the padding and border in an element's total width and height.

If you set box-sizing: border-box; on an element, padding and border are included in the width and height:

<!DOCTYPE html>

<html>

<head>

<style>

.div1 {

width: 300px;

height: 100px;

border: 1px solid blue;

box-sizing: border-box;

}

.div2 {

width: 300px;

height: 100px;

padding: 50px;

border: 1px solid red;

box-sizing: border-box;

}

</style>

</head>

<body>

<h1>With box-sizing</h1>

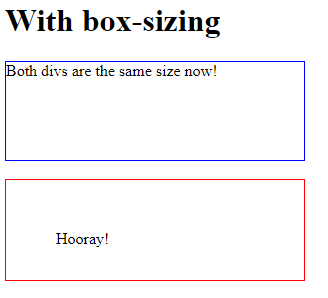
<div class="div1">Both divs are the same size now!</div>

<br>

<div class="div2">Hooray!</div>

</body>

</html>



## Definition and Usage

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

## Property Values

|  |  |  |
| --- | --- | --- |
| **Value** | **Description** | **Demo** |
| content-box | Default. The width and height properties (and min/max properties) includes only the content. Border and padding are not included | [Demo ❯](https://www.w3schools.com/cssref/playdemo.php?filename=playcss_box-sizing&preval=content-box) |
| border-box | The width and height properties (and min/max properties) includes content, padding and border |  |

<!DOCTYPE html>

<html>

<head>

<style>

#example1 {

box-sizing: content-box;

width: 300px;

height: 100px;

padding: 30px;

border: 10px solid blue;

}

#example2 {

box-sizing: border-box;

width: 300px;

height: 100px;

padding: 30px;

border: 10px solid blue;

}

</style>

</head>

<body>

<h1>The box-sizing Property</h1>

<p>Defines how the width and height of an element are calculated: should they include padding and borders, or not.</p>

<h2>box-sizing: content-box (default):</h2>

<p>Width and height only apply to the content of the element:</p>

<div id="example1">This div has a width of 300px. But the full width is 300px + 20px (left and right border) + 60px (left and right padding) = 380px!</div>

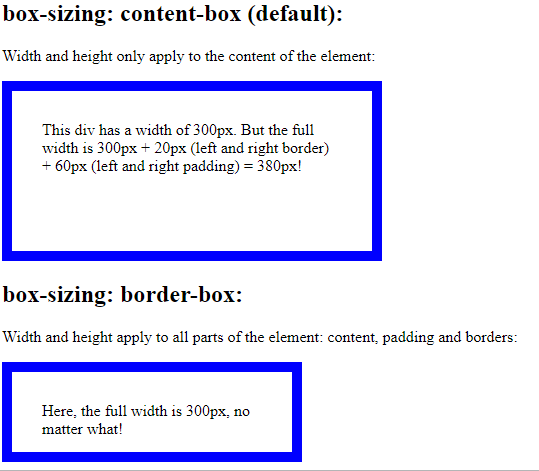
<h2>box-sizing: border-box:</h2>

<p>Width and height apply to all parts of the element: content, padding and borders:</p>

<div id="example2">Here, the full width is 300px, no matter what!</div>

</body>

</html>



Example

<!DOCTYPE html>

<html lang='en'>

<head>

<meta charset="utf-8">

<title>

How to create 3 boxes

in the same div using

HTML and CSS ?

</title>

<style>

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

margin: 2%;

justify-content: center;

overflow: hidden;

}

.box-wrapper {

height: 95vh;

width: 100%;

display: flex;

flex-direction: column;

text-align: center;

}

#box1 {

padding: 10px;

border: solid 1px green;

background-color:yellow;

}

#box2 {

padding: 8px;

border: solid 1px blue;

background-color:red;

}

#box3 {

padding: 10px;

flex-grow: 1;

display: flex;

flex-direction: row;

border: solid 1px green

}

#box4 {

flex-grow: 2;

border: solid 1px orange;

background-color:cyan;

}

.middle-column {

flex-grow: 1;

display: flex;

flex-direction: column;

background-color:green;

}

.middle-column div {

flex-grow: 1;

margin: 0 8px;

border: solid 1px red;

}

.middle-column div+div {

margin-top: 8px

}

#box8 {

flex-grow: 1;

border: solid 1px black;

background-color:blue;

}

</style>

</head>

<body>

<div class="box-wrapper">

<div id="box1">

Box 1

</div>

<div id="box2">

Box 2

</div>

<div id="box3">

<div id="box4">

Box 4

</div>

<div class="middle-column">

<div id="box5">

Box 5

</div>

<div id="box6">

Box 6

</div>

<div id="box7">

Box 7

</div>

</div>

<div id="box8">

Box 8

</div>

</div>

</div>

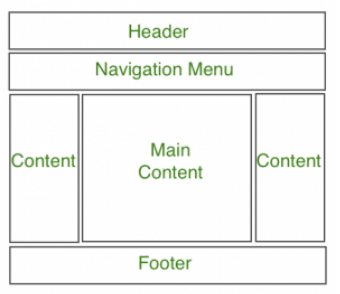
</body>

</html>



# CSS Website Layout

A website can be divided into various sections comprising of header, menus, content and footer based on which there are many different layout design available for developer. Different layouts can be created by using div tag and use CSS property to style it.



<!DOCTYPE html>

<html>

<head>

<title>

Website Layout

</title>

<style>

\* {

box-sizing: border-box;

}

/\* CSS property for header section \*/

.header {

background-color: cyan;

padding: 15px;

text-align: center;

}

/\* CSS property for navigation menu \*/

.nav\_menu {

overflow: hidden;

background-color: #333;

}

.nav\_menu a {

float: left;

display: block;

color: white;

text-align: center;

padding: 14px 16px;

text-decoration: none;

}

.nav\_menu a:hover {

background-color: white;

color: green;

}

/\* CSS property for content section \*/

.columnA, .columnB, .columnC {

float: left;

width: 31%;

padding: 15px;

text-align:justify;

}

h2 {

color:green;

text-align:center;

}

/\* Media query to set website layout

according to screen size \*/

@media screen and (max-width:600px) {

.columnA, .columnB, .columnC {

width: 50%;

}

}

@media screen and (max-width:400px) {

.columnA, .columnB, .columnC {

width: 100%;

}

}

.footer {

width: 100%;

bottom: 0px;

background-color: #000;

color: #fff;

position: absolute;

padding-top: 20px;

padding-bottom: 50px;

text-align: center;

font-size: 30px;

font-weight: bold;

}

</style>

</head>

<body>

<!-- header of website layout -->

<div class = "header">

<h2 style = "color:white;font-size:200%">

Graphic Era University

</h2>

</div>

<!-- navigation menu of website layout -->

<div class = "nav\_menu">

<a href = "#">Student</a>

<a href = "#">Teaching Staff</a>

<a href = "#">Non Teaching Staff</a>

</div>

<!-- Content section of website layout -->

<div class = "row">

<div class = "columnA">

<h2>VISION</h2>

<p>To impart quality education for producing highly talented globally recognizable technocrats

and entrepreneurs with sound ethics, latest knowledge and innovative ideas in Computer Science

and Engineering to meet industrial needs and societal expectations.</p>

</div>

<div class = "columnB">

<h2>Mission</h2>

<p>M1. To impart high standard value based technical education in all aspects of Computer

Science and Engineering through state of the art infrastructure and innovative approach.

M2. To produce ethical, motivated and skilled engineers through theoretical knowledge and practical applications.

M3. To impart the ability for tackling simple to complex problems individually as well as in a team..

M4. To develop globally competent engineers with strong foundations, capable of ―out of the box thinking so as

to adapt to the rapidly changing scenarios requiring social conscious green computing solutions.</p>

</div>

<div class = "columnC">

<h2>Program Outcomes</h2>

<p>Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an

engineering specialization to the solution of complex engineering problems.

Design/development of solutions: Design solutions for complex engineering problems and design system components

or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural,

societal, and environmental considerations.</p>

</div>

</div>

<div class="footer">

<p> Good Bye</p>

</div>

</body>

</html>

