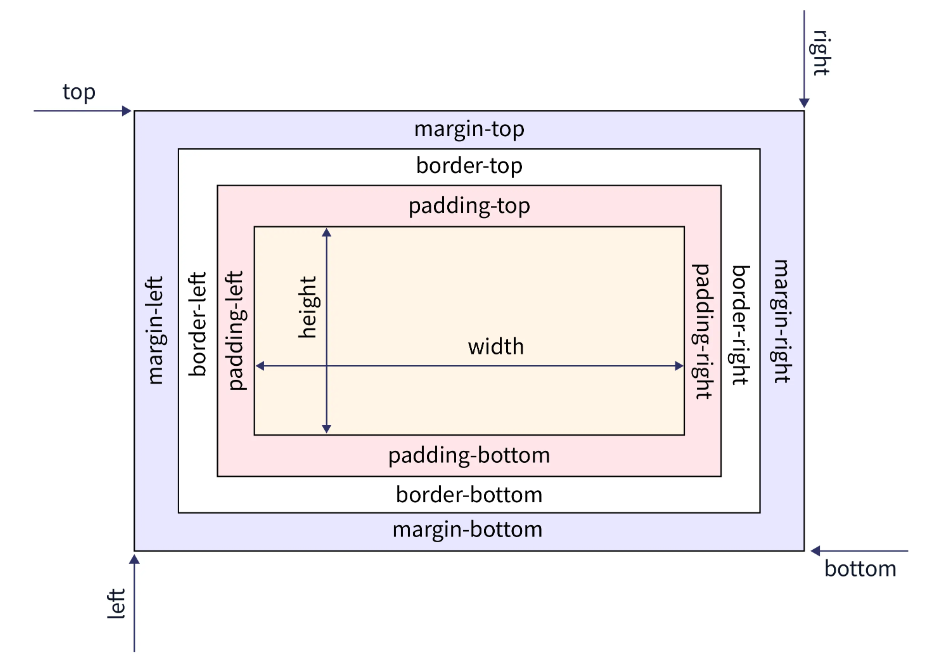
**What Is The CSS Box Model?**

The CSS Box Model describes all the HTML elements of the webpage as rectangular boxes.

A CSS basic box model consists of a content area, where any text, images, or other HTML elements are displayed. This is optionally surrounded by padding (space inside the element's border), a border (wraps the content area and padding), and a margin (space outside the element's border), on one or more sides. In the standard box model, if you give a box width and a height attribute, this defines the width and height of the content area of the element. Any padding and border is then added to that width and height to get the total size of the box.

**Diagram of The CSS Box Model:**

****

**Box-Model has multiple properties in CSS. Some of them are given below:**

* **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 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.

The total width for the element can be calculated as:

***Total element width = width + left padding + right padding + left border + right border + left margin + right margin***

The total height for the element can be calculated as:

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

**What is the Purpose of the CSS Box Model?**

The purpose of the CSS Box Model is to define all the elements on the webpage as a box. The browser uses the CSS box model to determine how an element should appear and how it should be positioned on a web page. It serves as a toolkit for positioning the elements, their content, and the elements surrounding them. The CSS Box Model, in a nutshell, is a layout guide for the elements on the webpage.

Padding Vs Margin

One can get easily confused between padding and margin as they both give the effect of adding space. The key differences between padding and margin are listed below.

|  |  |
| --- | --- |
| **Padding** | **Margin** |
| Padding is the inner space of an element i.e. the space inside the element’s border. | Margin is the outer space of an element i.e. the space outside the element’s border. |
| The styling of an element, such as the background color, affects the padding. | The styling of an element, such as the background color, does not affect the margin. |
| It does not allow negative values. | It can have negative values that draws the element closer to its neighbors than it would be by default. |

**How To Use the Padding Property in CSS**

The padding sits between the border and the content area. Unlike margins, padding cannot have negative values, so the value must be 0 or positive. Padding is commonly used to push the content away from the border. This property is a shorthand property for setting padding-top, padding-right, padding-bottom, and padding-left in a single declaration.

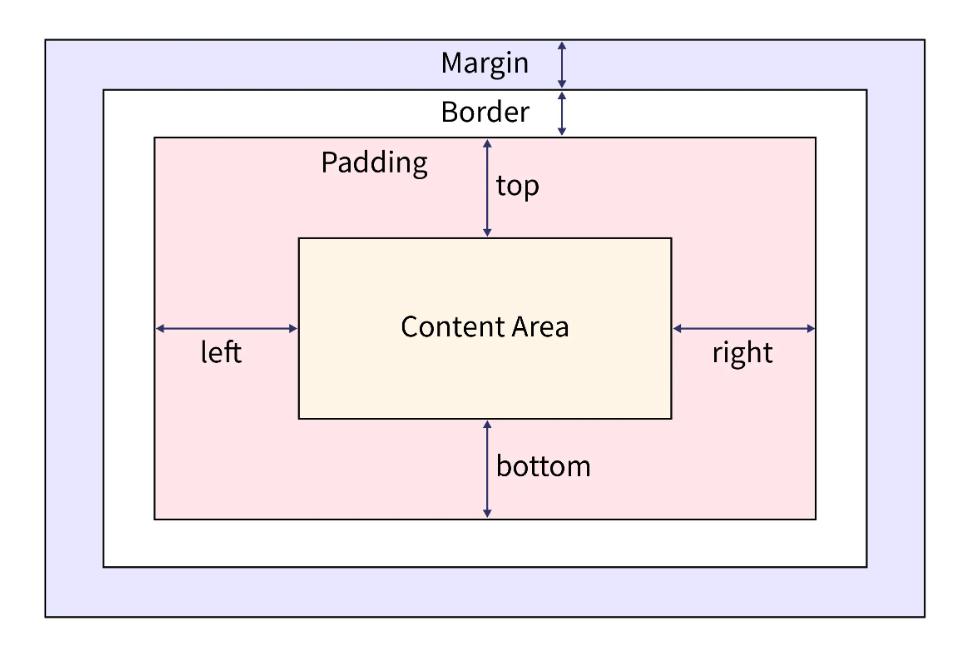
The padding property may be defined using one, two, three, or four values:

* 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, the third to the bottom.
* When four values are specified, the paddings apply to the top, right, bottom, and left.

Another approach to remember the order is by imagining a clock inside the box (an HTML element), and from the top, start moving clockwise i.e. TOP -> RIGHT -> BOTTOM -> LEFT For example, four values are given i.e. 5px, 8px, 3px, and 4px. Now, imagine how the hands would rotate on a clock: 5px starts in the 12 o'clock position, then 8px in the 3 o'clock position, then 3px in the 6 o'clock position, and 4px in the 9 o'clock position.

**Padding Properties:**

These are the following longhand padding properties of CSS, as shown in the diagram below:



* **padding-top:** The padding-top property of CSS sets the height of the padding area on the top of an element.
* **padding-right:** The padding-right property of CSS sets the width of the padding area on the right of an element.
* **padding-bottom:** The padding-bottom property of CSS sets the height of the padding area on the bottom of an element.
* **padding-left:** The padding-left property of CSS sets the width of the padding area on the left of an element.

The padding shorthand property of CSS sets the padding area on all four sides of an element in a single declaration. But the above longhand properties of padding allow us to specify the padding area for each side of an element.

**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 whatever 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.

**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, same as the padding property, and follows the same pattern. The values are either the 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, 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, 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.

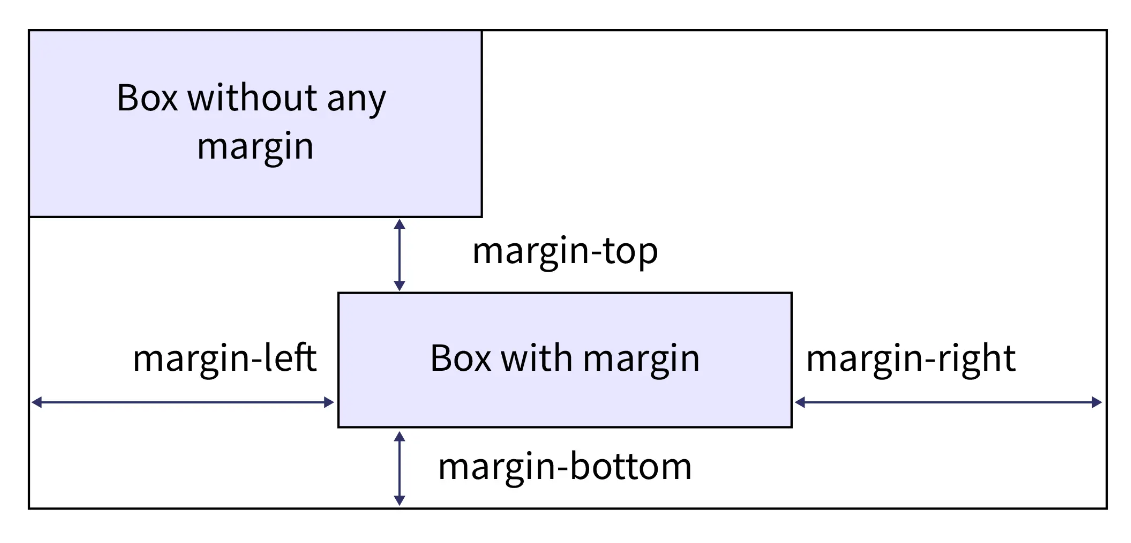
**Note: The default value of border-color property is currentColor if not defined by the user. It means that it inherits the value of color property of the element.**

**How to Use Margin Property in CSS:**

The margin property sets the margin area on all four sides of an element. They create extra space around an element. This margin propertsy is a shorthand property for setting margin on all four sides in a single declaration. The margin for each side of an element can be specified using following CSS properties:

* margin-top
* margin-right
* margin-bottom
* margin-left

In the margin attribute, we can put positive values, negative values, or an auto keyword. The auto keyword allows the browser to select a suitable margin. In certain cases, this value can be used to center an element within its container. And when we use negative values, it draws the element closer to its neighbors than it would be by default.



**Note:**

* **auto** keyword makes the element *horizontally centered*.
* When *negative* values are used, the element is drawn *closer* to its neighbours than it would be by default.

**What Is the Difference Between Content-Box and Border-Box in CSS?**

|  |  |
| --- | --- |
| **content-box** | **border-box** |
| Padding and borders are not included in an HTML element’s (it can be div, span, etc) width and height. | Padding and borders are included in an HTML element's (it can be div, span, etc) width and height. |
| By adding border and padding, the element's dimensions will change, i.e., they end up taller and wider than the specified dimensions. | By adding border and padding, the element's dimensions will remain the same as the specified dimensions, but the content area will shrink to absorb the extra width from padding and border. |
| This is the default value, it doesn't matter whether the user mentions it or not. | The user has to mention this value in the program. |

**What Values are Assigned to Box Model Properties?**

There are two types of units which can be assigned to box model properties i.e. relative and absolute. The following are all absolute length units, implying they are not relative to anything else and are considered to always be the same size.

* px: CSS pixels.
* cm: Centimeters.
* mm: Millimeters.

Relative length units are relative to something else, perhaps the size of the parent element's font, or the size of the viewport. Some of them are listed below.

* em: Font size of the parent, in the case of properties like font-size, and font size of the element itself, in the case of other properties like width.
* ex: The height of the character “x” also known as the font’s x-height.
* rem: The font size of the root element.
* vw: 1% of the viewport's width.
* vh: 1% of the viewport's height.
* vm: The smaller value of vw and vh.

We can also use the percentage(%), a CSS data type to represent a percentage value. It is often used to define a size as relative to an element's parent object. It represents a fraction of some other value such as 50%.