

- [CSS Properties Definitions](#)
 - [Box Model](#)
 - [CSS Basic Box Model](#)
 - [Content Area](#)
 - [Padding Area](#)
 - [Border Area](#)
 - [Margin Area](#)
 - [Box Sizing](#)

CSS Properties Definitions

Box Model

CSS Basic Box Model

When laying out a document, the browser's rendering engine represents each element as a rectangular box according to the standard **CSS basic box model** . CSS determines the size, position, and properties (color, background, border size, etc.) of these boxes.

Every box is composed of four parts (or *areas*), defined by their respective edges: the *content edge* , *padding edge* , *border edge* , and *margin edge* .

Content Area

The **content area** , bounded by the content edge, contains the "real" content of the element, such as text, an image, or a video player. Its dimensions are the *content width* (or *content-box width*) and the *content height* (or *content-box height*). It often has a background color or background image.

If the **box-sizing** property is set to **content-box** (default) and if the element is a block element, the content area's size can be explicitly defined with the **width**, **min-width**, **max-width**, **height**, **min-height**, and **max-height** properties.

Padding Area

The **padding area** , bounded by the padding edge, extends the content area to include the element's padding. Its dimensions are the *padding-box width* and the *padding-box height* .

The thickness of the padding is determined by the `padding-top`, `padding-right`, `padding-bottom`, `padding-left`, and shorthand `padding` properties.

Border Area

The **border area** , bounded by the border edge, extends the padding area to include the element's borders. Its dimensions are the *border-box width* and the *border-box height* .

The thickness of the borders are determined by the `border-width` and shorthand `border` properties. If the `box-sizing` property is set to `border-box`, the border area's size can be explicitly defined with the `width`, `min-width`, `max-width`, `height`, `min-height`, and `max-height` properties. When there is a background (`background-color` or `background-image`) set on a box, it extends to the outer edge of the border (i.e. extends underneath the border in z-ordering). This default behavior can be altered with the `background-clip` CSS property.

Margin Area

The **margin area** , bounded by the margin edge, extends the border area to include an empty area used to separate the element from its neighbors. Its dimensions are the *margin-box width* and the *margin-box height* .

The size of the margin area is determined by the `margin-top`, `margin-right`, `margin-bottom`, `margin-left`, and shorthand `margin` properties. When `margin collapsing` occurs, the margin area is not clearly defined since margins are shared between boxes.

Finally, note that for non-replaced inline elements, the amount of space taken up (the contribution to the height of the line) is determined by the `line-height` property, even though the borders and padding are still displayed around the content

Box Sizing

The **box-sizing** CSS property sets how the total width and height of an element is calculated.

By default in the **CSS box model**, the **width** and **height** you assign to an element is applied only to the element's content box. If the element has any border or padding, this is then added to the **width** and **height** to arrive at the size of the box that's rendered on the screen. This means that when you set **width** and **height**, you have to adjust the value you give to allow for any border or padding that may be added. For example, if you have four boxes with **width: 25%;**, if any has left or right padding or a left or right border, they will not by default fit on one line within the constraints of the parent container.

The **box-sizing** property can be used to adjust this behavior:

- **content-box** gives you the default CSS box-sizing behavior. If you set an element's width to 100 pixels, then the element's content box will be 100 pixels wide, and the width of any border or padding will be added to the final rendered width, making the element wider than 100px.
- **border-box** tells the browser to account for any border and padding in the values you specify for an element's width and height. If you set an element's width to 100 pixels, that 100 pixels will include any border or padding you added, and the content box will shrink to absorb that extra width. This typically makes it much easier to size elements. **box-sizing: border-box** is the default styling that browsers use for the `<table>`, `<select>`, and `<button>` elements, and for `<input>` elements whose type is `radio` , `checkbox` , `reset` , `button` , `submit` , `color` , or `search` .

Content Box

This is the initial and default value as specified by the CSS standard. The **width** and **height** properties include the content, but does not include the padding, border, or margin. For example, `.box {width: 350px; border: 10px solid black;}` renders a box that is 370px wide.

Here, the dimensions of the element are calculated as: *width = width of the content* , and *height = height of the content* . (Borders and padding are not included in the calculation.)

Border Box

The `width` and `height` properties include the content, padding, and border, but do not include the margin. Note that padding and border will be inside of the box. For example, `.box {width: 350px; border: 10px solid black;}` renders a box that is 350px wide, with the area for content being 330px wide. The content box can't be negative and is floored to 0, making it impossible to use `border-box` to make the element disappear.

Here the dimensions of the element are calculated as: $width = border + padding + width\ of\ the\ content$, and $height = border + padding + height\ of\ the\ content$.