

The CSS Box Model

For explaining how to use CSS to arrange objects (elements) we have understand the CSS Box Model. HTML elements can be considered as boxes. 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 HTML elements, and it consists of: margins, borders, padding, and the actual content.

The box model allows us to place a border around elements and space elements in relation to other elements.

The image below illustrates the box model:



Explanation of the different parts:

- **Margin** - Clears an area around the border. The margin does not have a background color, it is completely transparent
- **Border** - A border that goes around the padding and content. The border is affected by the background color of the box
- **Padding** - Clears an area around the content. The padding is affected by the background color of the box
- **Content** - The content of the box, where text and images appear

In order to set the width and height of an element correctly in all browsers, you need to know how the box model works.

Width and Height of an Element

Important: When you specify the width and height properties of an element with CSS, you are just setting the width and height of the content area. To know the full size of the element, you must also add the padding, border and margin.

The total width of the element in the example below is 300px:

```
width: 250px;  
padding: 10px;  
border: 5px solid gray;  
margin: 10px;
```

Let's do the math:

250px (width)
+ 20px (left and right padding)
+ 10px (left and right border)
+ 20px (left and right margin)
= 300px

Imagine that you only had 250px of space. Let's make an element with a total width of 250px:

Example

```
width: 220px;  
padding: 10px;  
border: 5px solid gray;  
margin: 0px;
```

The total width of an element should always be calculated like this:

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

The total height of an element should always be calculated like this:

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

Browsers Compatibility Issue

If you tested the previous example in Internet Explorer, you saw that the total width was not exactly 250px.

IE includes padding and border in the width, when the width property is set, **unless a DOCTYPE is declared.**

To fix this problem, just add a DOCTYPE to the code:

Example

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html>  
<head>  
<style type="text/css">  
div.ex  
{  
width: 220px;  
padding: 10px;  
border: 5px solid gray;  
margin: 0px;  
}  
</style>  
</head>
```

Border Style

The border-style property specifies what kind of border to display.

Note: None of the border properties will have ANY effect unless the border-style property is set!

border-style values:

none: Defines no border

dotted: Defines a dotted border

dashed: Defines a dashed border

solid: Defines a solid border

double: Defines two borders. The width of the two borders are the same as the border-width value

groove: Defines a 3D grooved border. The effect depends on the border-color value

ridge: Defines a 3D ridged border. The effect depends on the border-color value

inset: Defines a 3D inset border. The effect depends on the border-color value

outset: Defines a 3D outset border. The effect depends on the border-color value

Border Width

The border-width property is used to set the width of the border.

The width is set in pixels, or by using one of the three pre-defined values: thin, medium, or thick.

Note: The "border-width" property does not work if it is used alone. Use the "border-style" property to set the borders first.

Example

```
p.one
{
border-style:solid;
border-width:5px;
}
p.two
{
border-style:solid;
border-width:medium;
}
```

Border Color

The border-color property is used to set the color of the border. The color can be set by:

- name - specify a color name, like "red"
- RGB - specify a RGB value, like "rgb(255,0,0)"
- Hex - specify a hex value, like "#ff0000"

You can also set the border color to "transparent".

Note: The "border-color" property does not work if it is used alone. Use the "border-style" property to set the borders first.

Example

```
p.one
{
border-style:solid;
border-color:red;
}
p.two
{
border-style:solid;
border-color:#98bf21;
}
```

Border - Individual sides

In CSS it is possible to specify different borders for different sides:

Example

```
p
{
border-top-style:dotted;
border-right-style:solid;
border-bottom-style:dotted;
border-left-style:solid;
}
```

The example above can also be set with a single property:

Example

```
border-style:dotted solid;
```

The border-style property can have from one to four values.

- **border-style:dotted solid double dashed;**
 - top border is dotted
 - right border is solid
 - bottom border is double
 - left border is dashed
- **border-style:dotted solid double;**
 - top border is dotted
 - right and left borders are solid
 - bottom border is double
- **border-style:dotted solid;**

- top and bottom borders are dotted
- right and left borders are solid
- **border-style:dotted;**
 - all four borders are dotted

The border-style property is used in the example above. However, it also works with border-width and border-color.

CSS Outlines

An outline is a line that is drawn around elements, outside the border edge, to make the element "stand out".

The outline properties specifies the style, color, and width of an outline.

Example

```
border:1px solid red;
outline-style:dotted;
outline-width:thick;
outline-color:green;
```

CSS Margin

The CSS margin properties define the space around elements.

Margin

The margin clears an area around an element (outside the border). The margin does not have a background color, and is completely transparent.

The top, right, bottom, and left margin can be changed independently using separate properties. A shorthand margin property can also be used, to change all margins at once.

Possible Values

Value	Description
Auto	The browser sets the margin. The result of this is dependant of the browser
<i>length</i>	Defines a fixed margin (in pixels, pt, em, etc.)
%	Defines a margin in % of the containing element

Note: It is possible to use negative values, to overlap content.

Margin - Individual sides

In CSS, it is possible to specify different margins for different sides:

Example

```
margin-top: 100px;  
margin-bottom: 100px;  
margin-right: 50px;  
margin-left: 50px;
```

Margin - Shorthand property

To shorten the code, it is possible to specify all the margin properties in one property. This is called a shorthand property.

The shorthand property for all the margin properties is "margin":

Example

```
margin: 100px 50px;
```

The margin property can have from one to four values.

- **margin:25px 50px 75px 100px;**
 - top margin is 25px
 - right margin is 50px
 - bottom margin is 75px
 - left margin is 100px
- **margin:25px 50px 75px;**
 - top margin is 25px
 - right and left margins are 50px
 - bottom margin is 75px
- **margin:25px 50px;**
 - top and bottom margins are 25px
 - right and left margins are 50px
- **margin:25px;**
 - all four margins are 25px

CSS Padding

The CSS padding properties define the space between the element border and the element content.

Padding

The padding clears an area around the content (inside the border) of an element. The padding is affected by the background color of the element. The top, right, bottom, and left padding can be changed independently using separate properties. A shorthand padding property can also be used, to change all paddings at once.

Possible Values

Value	Description
<i>length</i>	Defines a fixed padding (in pixels, pt, em, etc.)
%	Defines a padding in % of the containing element

Padding - Individual sides

In CSS, it is possible to specify different padding for different sides:

Example

```
padding-top: 25px;  
padding-bottom: 25px;  
padding-right: 50px;  
padding-left: 50px;
```

Padding - Shorthand property

To shorten the code, it is possible to specify all the padding properties in one property. This is called a shorthand property.

The shorthand property for all the padding properties is "padding":

Example

```
padding: 25px 50px;
```

The padding property can have from one to four values.

- **padding:25px 50px 75px 100px;**
 - top padding is 25px
 - right padding is 50px
 - bottom padding is 75px
 - left padding is 100px
- **padding:25px 50px 75px;**
 - top padding is 25px
 - right and left paddings are 50px
 - bottom padding is 75px
- **padding:25px 50px;**
 - top and bottom paddings are 25px
 - right and left paddings are 50px
- **padding:25px;**
 - all four paddings are 25px

Borders

```
<html>
<head>
<style type="text/css">
p.none {border-style:none;}
p.dotted {border-style:dotted;}
p.dashed {border-style:dashed;}
p.solid {border-style:solid;}
p.double {border-style:double;}
p.groove {border-style:groove; border-color:green;}
p.ridge {border-style:ridge; border-color:green;}
p.inset {border-style:inset; border-color:green;}
p.outset {border-style:outset; border-color:green;}
p.hidden {border-style:hidden; border-color:green;}
</style>
</head>

<body>
<p class="none">No border.</p>
<p class="dotted">A dotted border.</p>
<p class="dashed">A dashed border.</p>
<p class="solid">A solid border.</p>
<p class="double">A double border.</p>
<p class="groove">A groove border.</p>
<p class="ridge">A ridge border.</p>
<p class="inset">An inset border.</p>
<p class="outset">An outset border.</p>
<p class="hidden">A hidden border.</p>
</body>

</html>
```


Border color

```
<html>
<head>
<style type="text/css">
p.one
{
border-style:solid;
border-color:red;
}
p.two
{
border-style:solid;
border-width:100px;
border-color:transparent;
}
p.three
{
border-style:solid;
border-color:#98bf21;
}
</style>
</head>

<body>
<p class="one">A solid red border</p>
<p class="two">A solid transparent border</p>
<p class="three">A solid green border</p>
<p><b>Note:</b> The "border-color" property does not work if it is used alone.
Use the "border-style" property to set the borders first.</p>
</body>
</html>
```

Border Individual sides

```
<html>
<head>
<style type="text/css">
p
{
border-top-style:dotted;
border-right-style:solid;
border-bottom-style:dotted;
border-left-style:solid;
}
</style>
</head>

<body>
<p>2 different border styles.</p>
</body>
</html>
```

Outlines

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
<style type="text/css">
p
{
border:1px solid red;
outline:green dotted 5px;
}
h3
{
border:1px solid red;
outline-style:dotted;
outline-width:thick;
outline-color:green;
}
</style>
</head>
<body>
<p><b>Note:</b> Internet Explorer 8 (and higher) supports the outline property
if a !DOCTYPE is specified.</p>
<h3>Note: Internet Explorer 8 (and higher) supports the outline property
if a !DOCTYPE is specified.</h3>
</body>
</html>
```

Margins

```
<html>
<head>
<style type="text/css">
p{
background-color:yellow;
}
p.margin{
margin-top:100px;
margin-bottom:100px;
margin-right:50px;
margin-left:50px;
}
</style>
</head>
<body>
<p>This is a paragraph with no specified margins.</p>
<p class="margin">This is a paragraph with specified margins.</p>
</body>
</html>
```

Padding

```
<html>
<head>
<style type="text/css">
p
{
background-color:yellow;
}
p.padding
{
padding-top:25px;
padding-bottom:25px;
padding-right:50px;
padding-left:50px;
}
</style>
</head>

<body>
<p>This is a paragraph with no specified padding.</p>
<p class="padding">This is a paragraph with specified paddings.</p>
</body>

</html>
```

Source: w3schools.com
