# CSS: position property

This CSS tutorial explains how to use the CSS property called **position** with syntax and examples.

## Description

The CSS position property defines the position of an element where generally the top, right, bottom, and left properties will determine the position of the element.

## Syntax

The syntax for the position CSS property is:

position: value;

### **Parameters or Arguments**

**value**

The rule to be used to position the element. It can be one of the following:

| Value | Description |
| --- | --- |
| static | Normal position for the element (where top, right, bottom, and left have no effect) div { position: static; } |
| relative | Position the element relative to where its normal position would have been div { position: relative; top: 10px; left: 15px; } |
| absolute | Position the element absolutely relative to its container div { position: absolute; top: 10px; left: 15px; } |
| fixed | Position the element relative to the screen's viewport and stay fixed on screen when scrolling div { position: fixed; top: 10px; left: 15px; } |
| inherit | Indicates that the element will inherit the position from its parent element div { position: inherit; } |

## Note

* If the position property is set to relative, absolute, or fixed, you can set the top, right, bottom, and left properties.
* If the position property is set to static, he top, right, bottom, and left properties do not apply.
* See also the [top](https://www.techonthenet.com/css/properties/top.php), [right](https://www.techonthenet.com/css/properties/right.php), [bottom](https://www.techonthenet.com/css/properties/bottom.php), and [left](https://www.techonthenet.com/css/properties/left.php) properties.

## Browser Compatibility

The CSS position property has basic support with the following browsers:

* Chrome
* Firefox (Gecko)
* Internet Explorer (IE)
  + static, relative, absolute: IE 4
  + fixed: IE 7
* Opera
* Safari (WebKit)

## Example

We will discuss the position property below, exploring examples of how to use this property in CSS.

### **Static Position**

Let's look at an example where we set the position to static.

The CSS would look like this:

.totn\_container { background: red; padding: 10px; width: 360px; height: 40px; }

.totn1, .totn2, .totn3 { float: left; background: lightgreen; border: 2px solid blue; width: 115px; }

.totn2 { position: static; }

The HTML would look like this:

<div class="totn\_container">

<div class="totn1">Div 1 - techonthenet.com</div>

<div class="totn2">Div 2 - techonthenet.com</div>

<div class="totn3">Div 3 - techonthenet.com</div>

</div>

The result would look like this:

CSS

In this CSS position example, we have created 3 divs that are set to [float](https://www.techonthenet.com/css/properties/float.php) left and are contained within an outer div with a class called totn\_container. Each of the 3 divs is assigned a class - totn1, totn2, totn3, respectively.

Then we have set the totn2 class (ie: the second div) to have a position of static. Since this is the default position behavior, our result looks the same as if we had not specified a position value. In other words, it is the equivalent to this CSS:

.totn\_container { background: red; padding: 10px; width: 360px; height: 40px; }

.totn1, .totn2, .totn3 { float: left; background: lightgreen; border: 2px solid blue; width: 115px; }

### **Relative Position**

Let's take a look at what would happen if we tried setting the position property to relative for the second div.

The CSS would look like this:

.totn\_container { background: red; padding: 10px; width: 360px; height: 40px; }

.totn1, .totn2, .totn3 { float: left; background: lightgreen; border: 2px solid blue; width: 115px; }

.totn2 { position: relative; top: 20px; left: 15px; }

The HTML would look like this:

<div class="totn\_container">

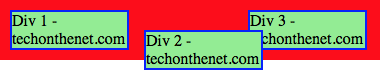
<div class="totn1">Div 1 - techonthenet.com</div>

<div class="totn2">Div 2 - techonthenet.com</div>

<div class="totn3">Div 3 - techonthenet.com</div>

</div>

The result would look like this:



In this CSS position example, we have set the totn2 class to have a position of relative and the top property is set to 20px and the left property is set to 15px. What this means is that the div element (identified by the totn2 class) will be moved down by 20px and moved left by 15px relative to where it would have been positioned in a normal layout (as seen in the first static example).

### **Absolute Position**

Let's take a look at what would happen if we tried setting the position property to absolute for the second div.

The CSS would look like this:

.totn\_container { background: red; padding: 10px; width: 360px; height: 40px; }

.totn1, .totn2, .totn3 { float: left; background: lightgreen; border: 2px solid blue; width: 115px; }

.totn2 { position: absolute; top: 20px; left: 15px; }

The HTML would look like this:

<div class="totn\_container">

<div class="totn1">Div 1 - techonthenet.com</div>

<div class="totn2">Div 2 - techonthenet.com</div>

<div class="totn3">Div 3 - techonthenet.com</div>

</div>

The result would look like this:

CSS

In this CSS position example, we have set the totn2 class to have a position of absolute and the top property is set to 20px and the left property is set to 15px. What this means is that the div element (identified by the totn2 class) will be moved within its parent container to an absolute position of 20px from the top of the container and 15px from the left of the container.

So in this example, the second div in effect is positioned on top of the first div (identified by the totn1 class).

### **Fixed Position**

Let's take a look at what would happen if we tried setting the position property to fixed for the second div.

The CSS would look like this:

.totn\_container { background: red; padding: 10px; width: 360px; height: 40px; }

.totn1, .totn2, .totn3 { float: left; background: lightgreen; border: 2px solid blue; width: 115px; }

.totn2 { position: fixed; top: 20px; left: 15px; }

The HTML would look like this:

<div class="totn\_container">

<div class="totn1">Div 1 - techonthenet.com</div>

<div class="totn2">Div 2 - techonthenet.com</div>

<div class="totn3">Div 3 - techonthenet.com</div>

</div>

The result would look like this:

CSS

In this CSS position example, we have set the totn2 class to have a position of fixed and the top property is set to 20px and the left property is set to 15px. What this means is that the div element (identified by the totn2 class) will be moved within the screen's viewport to a fixed position of 20px from the top of the viewport and 15px from the left of the viewport.

It looks very similar to the result from the absolute position example. The only exception is that when we scroll with the fixed position example, the second div (identified by the totn2 class) will remain fixed within the screen's viewport and will not scroll off of the screen.