

<https://leannezhang.medium.com/difference-between-css-position-absolute-versus-relative-35f064384c6>

Leanne Zhang

Follow

Jan 9, 2017 · 3 min read · Listen



Difference between css position absolute versus relative

I spent sometime understanding the difference between absolute positioning and relative positioning. It was a confusing topic to me and I decide to illustrate their differences with pictures.

Before going further, we should know the default behaviour of position when we don't specify any position property.

position: static

By default, position an element based on its current position in the flow. The `top`, `right`, `bottom`, `left` and `z-index` properties do not apply. — source MDN

```
position: relative
```

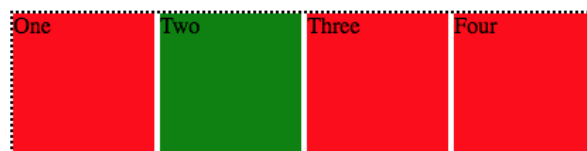
Position an element based on its current position without changing layout.

```
position: absolute
```

Position an element based on its closest positioned ancestor position.

Take an example

To start with, create a parent container with 4 boxes side by side.



```
position: static
```

[Sign In](#)[Get started](#)

index.html

```
<div class="parent">
  <div class="box" id="one">One</div>
  <div class="box" id="two">Two</div>
  <div class="box" id="three">Three</div>
  <div class="box" id="four">Four</div>
</div>
```

style.css

```
.parent {
  border: 2px black dotted;
  display: inline-block;
}

.box {
  display: inline-block;
  background: red;
  width: 100px;
  height: 100px;
}

#two {
  background: green;
}
```

Live code: [JsBin](#)

By default, position is set to static. It positions based on the layout in the flow.

What happens when we want to move the GreenBox but do not want to affect the layout around it?



position: relative

This is where position relative comes in. Move the green box relative to its current position to 20px from the left and top without changing the layout around it. Thus, **leaving a gap** for the green box where it would have been had it not been position.



Sign In

Get started

```
#two {
  top: 20px;
  left: 20px;
  background: green;
  position: relative;
}
```

Live code: [JsBin](#)

Position: absolute is the opposite.



position: absolute

By applying `position: absolute` to the GreenBox, it will **not** leave any gap where it would have been. The position of the GreenBox is based on its parent position (the dotted border). Thus, moving 20px to the left and bottom from the top-left origin of the dotted border.

```
#two {
  top: 20px;
  left: 20px;
  background: green;
  position: absolute;
}
```

Live code: [JsBin](#)

In a nutshell ...

`position: relative` places an element relative to its current position without changing the layout around it, whereas `position: absolute` places an element relative to its parent's position and changing the layout around it.

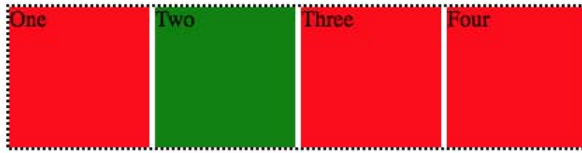




Sign In

Get started

position: static



position: relative



position: absolute



Medium

About Help Terms Privacy

