





# Overview

CSS Positioning and Layout

## What does CSS position do?

# CSS POSITION PROPERTY

Using CSS, you can layout all your elements on your webpage visually. For example, you can position an element at the very top of your page, or 50px below the element before it.

To control just how an element will appear in the layout, you need to use the CSS position property.

# What is the CSS position property?

## CSS POSITION PROPERTY

- ❑ specifies the type of positioning method used for an element.
- ❑ Elements are then positioned using the top, bottom, left, and right properties.
- ❑ However, these properties will not work unless the position property is set first.
- ❑ They also work differently depending on the position value.
  
- ❑ 5 different position values:
  1. Static
  2. Relative
  3. Absolute
  4. Fixed
  5. sticky

# position: static;

- ❑ HTML elements are positioned static by default.
- ❑ Static positioned elements are not affected by the top, bottom, left, and right properties.
- ❑ An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page:

Example:

This <div> element has position: static;

Here is the CSS that is used:

```
div.static {  
  position: static;  
  border: 3px solid #73AD21;  
}
```

# position: relative;

- ❑ An element with `position: relative;` is positioned relative to its normal position.
- ❑ Setting the `top`, `right`, `bottom`, and `left` properties of a relatively-positioned element will cause it to be adjusted away from its normal position.
- ❑ Other content will not be adjusted to fit into any gap left by the element.

Example:

This `<div>` element has `position: relative;`

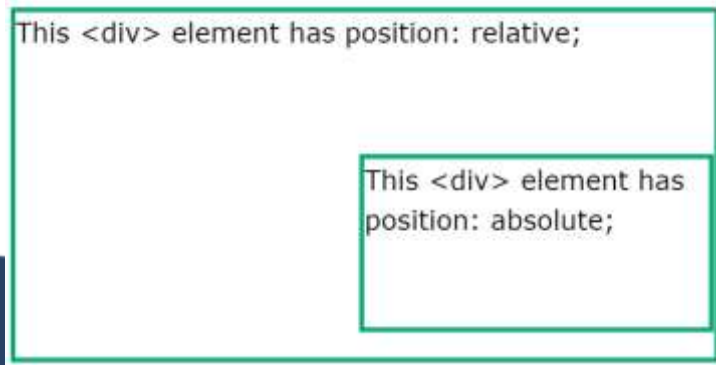
Here is the CSS that is used:

```
div.relative {  
    position: relative;  
    left: 30px;  
    border: 3px solid #73AD21;  
}
```

# position: absolute;

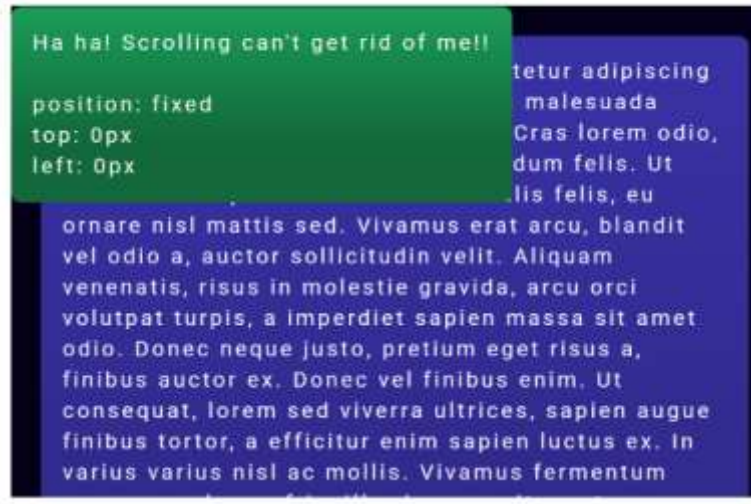
- ❑ Absolute positioned elements are removed from the normal flow, and can overlap elements.
- ❑ will cause an element to be taken out of that normal flow of the webpage.
- ❑ An absolutely positioned element needs to position itself in relation to a positioned ancestor

Example:



# position: fixed;

- ❑ An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled.
- ❑ The top, right, bottom, and left properties are used to position the element.
- ❑ A fixed element does not leave a gap in the page where it would normally have been located.

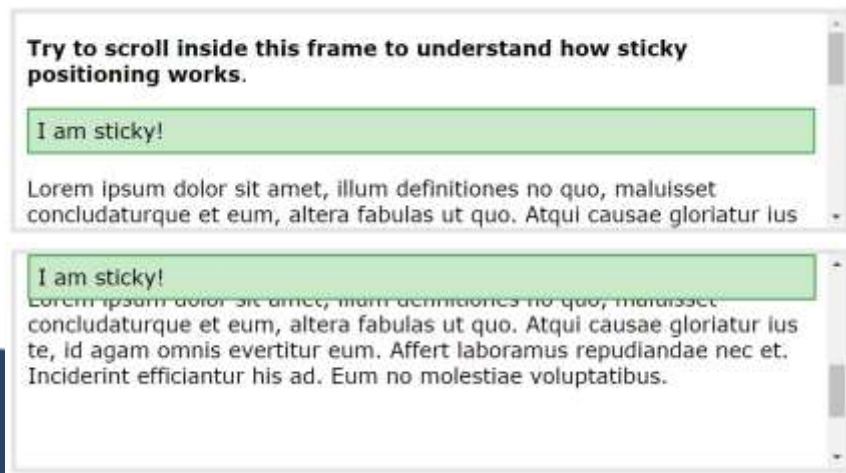




# position: sticky;

- ❑ An element with position: sticky; is positioned based on the user's scroll position.
- ❑ A sticky element toggles between relative and fixed, depending on the scroll position.
- ❑ Will initially behave like position: relative elements, but if you keep scrolling, they will get taken out of the normal flow and behave like position: fixed.

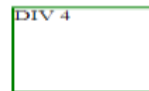
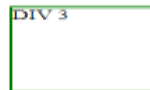
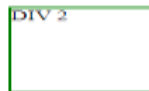
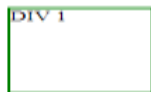
Example:



# Exercise

Create a webpage consist of the following.

1.



2.

