CSS Notes For Students By Geekster

Geekster

www.geekster.in

Positions in CSS

The **`position`** property specifies the type of positioning method used for an element (static, relative, fixed, absolute or sticky).

The position Property

The **`position`** property specifies the type of positioning method used for an element.

There are five different position values:

- `static`
- `relative`
- `fixed`
- `absolute`
- `sticky`

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.

position: static;

HTML elements are positioned static by default.

Static positioned elements are not affected by the top, bottom, left, and right properties.

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.

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: absolute;

An element with `position: absolute;` is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.

Note: Absolute positioned elements are removed from the normal flow, and can overlap elements.

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. It is positioned relative until a given offset position is met in the viewport - then it "sticks" in place (like position:fixed).

Note: Internet Explorer does not support sticky positioning. Safari requires a -webkit- prefix (see example below). You must also specify at least one of `top`, `right`, `bottom` or `left` for sticky positioning to work

Designing a Sticky Navbar

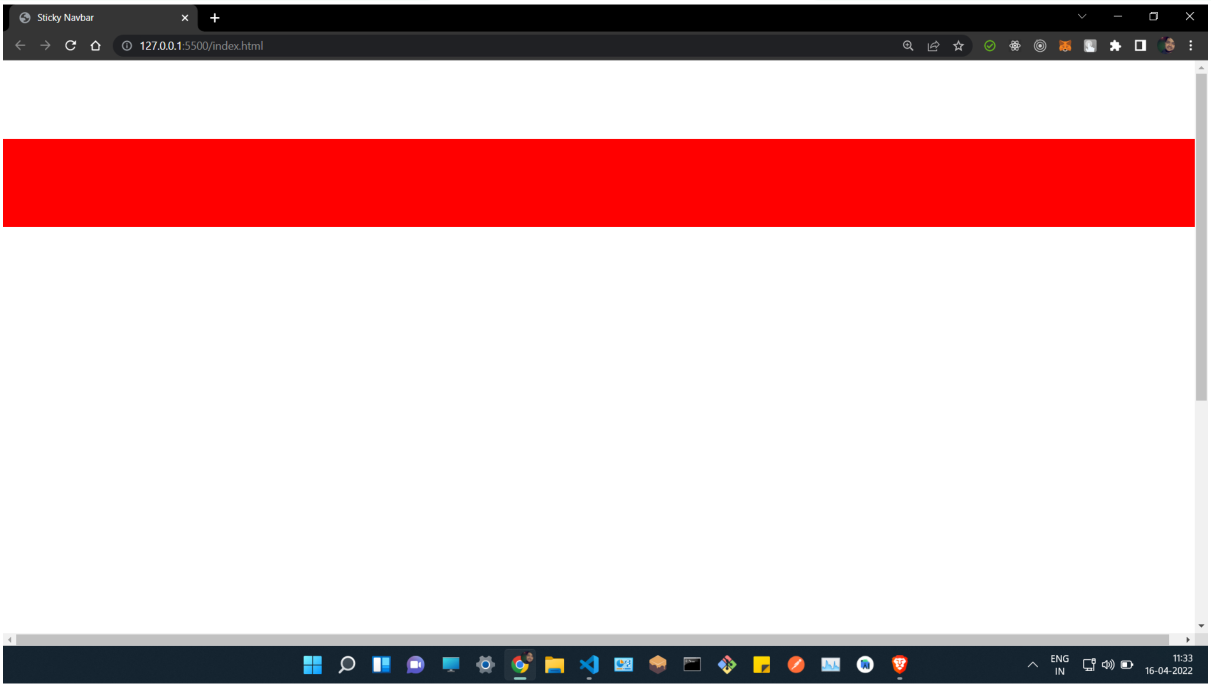
1. Define a navbar in the body.

```
<body>
    <nav class="sticky" >
    </nav>
</body>
```

2. Define the css of the page.

```
        *{
            margin: 0;
            height: 150vh;
        }
        nav.sticky{
            height:15vh;
            width: 100vw;
            background-color: red;
            margin-top:  5rem; ;
        }
```

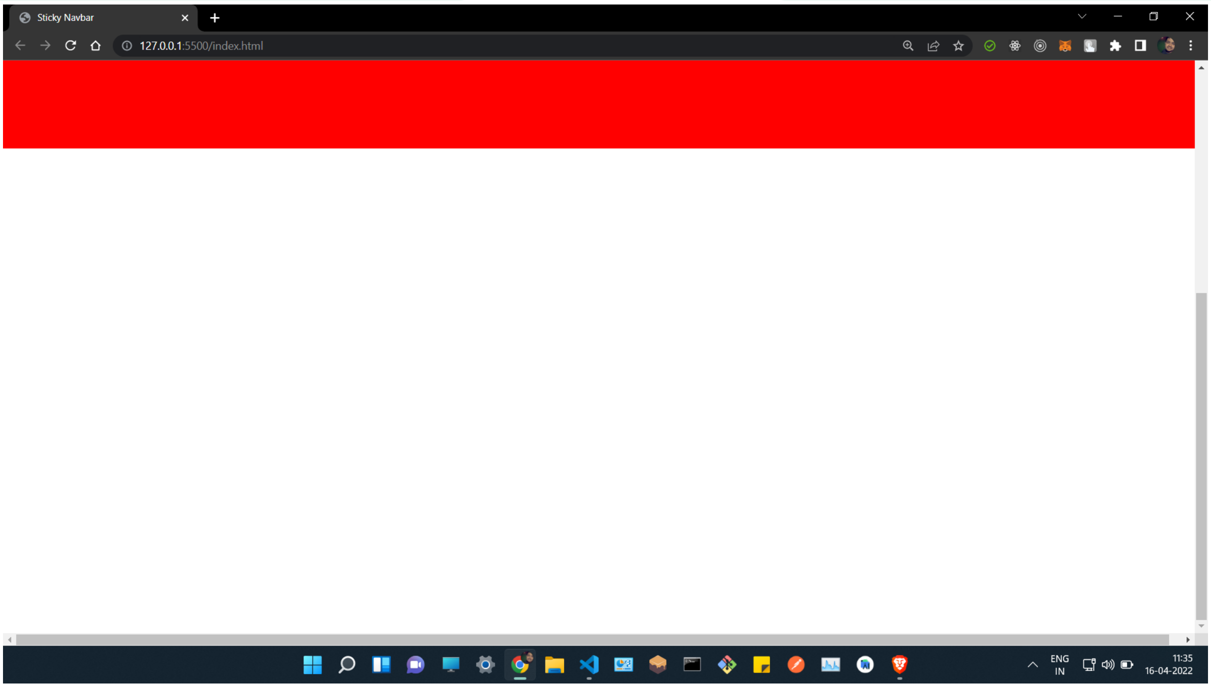
3. Web page till now, It does not have sticky functionality, so on Scroll It will disappear.



4. Now add the position and from where you have to make it sticky.

```
        nav.sticky{
            height:15vh;
            width: 100vw;
            background-color: red;
            margin-top:  5rem;
            position: sticky;
            top: 0;
        }
```

5. Now On scrolling, It is sticky.



1. https://www.w3schools.com/css/css_positioning.asp
2. https://www.w3schools.com/cssref/pr_class_position.asp
3. https://developer.mozilla.org/en-US/docs/Web/CSS/position
4. https://css-tricks.com/almanac/properties/p/position/
5. https://www.youtube.com/watch?v=MwGHiVl-gqk
6. https://www.youtube.com/watch?v=gD3G67oPg-w