CSS | Positioning Elements

The *position* property in CSS tells about the method of positioning for an element or an HTML entity. There are five different types of position property available in CSS:

1. Fixed
2. Static
3. Relative
4. Absolute
5. Sticky

The positioning of an element can be done using the *top*, *right*, *bottom* and *left* property. These specify the distance of an HTML element from the edge of the viewport. To set the position by these four properties, we have to declare the positioning method.
Let's talk about each of these position methods in details:

1. Fixed

Any HTML element with **position: fixed** property will be positioned relative to the viewport. An element with fixed positioning allows it to remain at the same position even we scroll the page. We can set the position of the element using the top, right, bottom, left.

```
<!-->html code<-->

<body>

    <div class="fixed">This div has <span>position: fixed;</span></div>

    <pre>

        Lorem ipsum dolor sits amet, consectetur adipiscing elit.

        Nunc eget mauris at urna hendrerit iaculis sit amet et ipsum.

        Maecenas nec mi eget leo malesuada vehicula.

        Nam eget velit maximus, elementum ante pretium, aliquet felis.

        Aliquam quis turpis laoreet, porttitor lacus at, posuere massa.
```

```
    </pre>
</body>
```

Below is the CSS code to illustrate the fixed property:

```
// css code
body
{
    margin: 0;
    padding: 20px;
    font-family: sans-serif;
    background: #efefef;
}

.fixed
{
    position: fixed;
    background: #cc0000;
    color: #ffffff;
    padding: 30px;
    top: 50;
    left: 10;
}
```

```
span
{
   padding: 5px;
   border: 1px #ffffff dotted;
}
```

2. Static

This method of positioning is set by default. If we don't mention the method of positioning for any element, the element has the **position:static** method by default. By defining Static, the top, right, bottom and left will not have any control over the element. The element will be positioned with the normal flow of the page.

```
<!-->html code<-->
<body>
   <div class="static">This div has <span>position: static;</span></div>
   <pre>
      Lorem ipsum dolor sits amet, consectetur adipiscing elit.
      Nunc eget mauris at urna hendrerit iaculis sit amet et ipsum.
      Maecenas nec mi eget leo malesuada vehicula.
      Nam eget velit maximus, elementum ante pretium, aliquet felis.
      Aliquam quis turpis laoreet, porttitor lacus at, posuere massa.
   </pre>
</body>
```

Below is the CSS code to illustrate the static property:

```
// css code
body
```

```
{
   margin: 0;
   padding: 20px;
   font-family: sans-serif;
   background: #efefef;
}

.static
{
   position: static;
   background: #cc0000;
   color: #ffffff;
   padding: 30px;
}

span
{
   padding: 5px;
   border: 1px #ffffff dotted;
}
```

3. Relative

An element with **position: relative** is positioned relatively with the other elements which are sitting at top of it. If we set its top, right, bottom or left, other elements will not fill up the gap left by this element.

```
<!-->html code<-->
<body>
    <div class="relative">This div has
        <span>position: relative;</span></div>
    <pre>
        Lorem ipsum dolor sits amet, consectetur adipiscing elit.
        Nunc eget mauris at urna hendrerit iaculis sit amet et ipsum.
        Maecenas nec mi eget leo malesuada vehicula.
        Nam eget velit maximus, elementum ante pretium, aliquet felis.
        Aliquam quis turpis laoreet, porttitor lacus at, posuere massa.
    </pre>
</body>
```

Below is the CSS code to illustrate the relative property:

```
// css code
body
{
   margin: 0;
   padding: 20px;
   font-family: sans-serif;
   background: #efefef;
}

.relative
```

```
{
   position: relative;
   background: #cc0000;
   color: #ffffff;
   padding: 30px;
}


span
{
   padding: 5px;
   border: 1px #ffffff dotted;
}
```

4. Absolute

An element with **position: absolute** will be positioned with respect to its parent. Positioning of this element does not depend upon its siblings or the elements which are at same level.

filter_none

brightness_4

```
<!-->html code<-->

<body>
    <pre>
       Lorem ipsum dolor sits amet, consectetur adipiscing elit.
       Nunc eget mauris at urna hendrerit iaculis sit amet et ipsum.
       Maecenas nec mi eget leo malesuada vehicula.
       <div class="relative">
```

```
        <p>This div has <span><strong>position: relative;</strong>
                                        </span></p>
        <div class="absolute">
          This div has <span><strong>position:
                    absolute;</strong></span>
        </div>
      </div>
      Nam eget velit maximus, elementum ante pretium, aliquet felis.
      Aliquam quis turpis laoreet, porttitor lacus at, posuere massa.
    </pre>
</body>
```

Below is the CSS code to illustrate the absolute property:

```
// css code
body
{
   margin: 0;
   padding: 20px;
   font-family: sans-serif;
   background: #efefef;
}

.absolute
{
```

```
    position: absolute ;
    background: #cc0000;
    color: #ffffff;
    padding: 30px;
    font-size: 15px;
    bottom: 20px;
    right: 20px;
}

.relative
{
    position: relative;
    background: #aad000;
    height: 300px;
    font-size: 30px;
    border: 1px solid #121212;
    text-align: center;
}

span
{
    padding: 5px;
    border: 1px #ffffff dotted;
}
```

```
pre
{
    padding: 20px;
    border: 1px solid #000000;
}
```

5. Sticky

Element with **position: sticky** and **top: 0** played a role between **fixed & relative** based on the position where it is placed. If the element is placed at the middle of the document then when user scrolls the document, the sticky element starts scrolling until it touch the top. When it touches the top, it will be fixed at that place inspite of further scrolling. We can stick the element at bottom, with the **bottom** property.

```
<!-->html code<-->
<body>
    <pre>
        Lorem ipsum dolor sits amet, consectetur adipiscing elit.
        Nunc eget mauris at urna hendrerit iaculis sit amet et ipsum.
        Maecenas nec mi eget leo malesuada vehicula.
            <div class="sticky">
                This div has <span>position: sticky;</span>
            </div>
        Nam eget velit maximus, elementum ante pretium, aliquet felis.
        Aliquam quis turpis laoreet, porttitor lacus at, posuere massa.
    </pre>
```

</body>

Below is the CSS code to illustrate the sticky property:

```
// css code
body
{
   margin: 0;
   padding: 20px;
   font-family: sans-serif;
   background: #efefef;
}

.sticky
{
   position: sticky;
   background: #cc0000;
   color: #ffffff;
   padding: 30px;
      top: 10px;
      right: 50px;
}

span
{
```

```
    padding: 5px;
    border: 1px #ffffff dotted;
}

pre
{
      padding: 20px;
      border: 1px solid #000000;
}
```