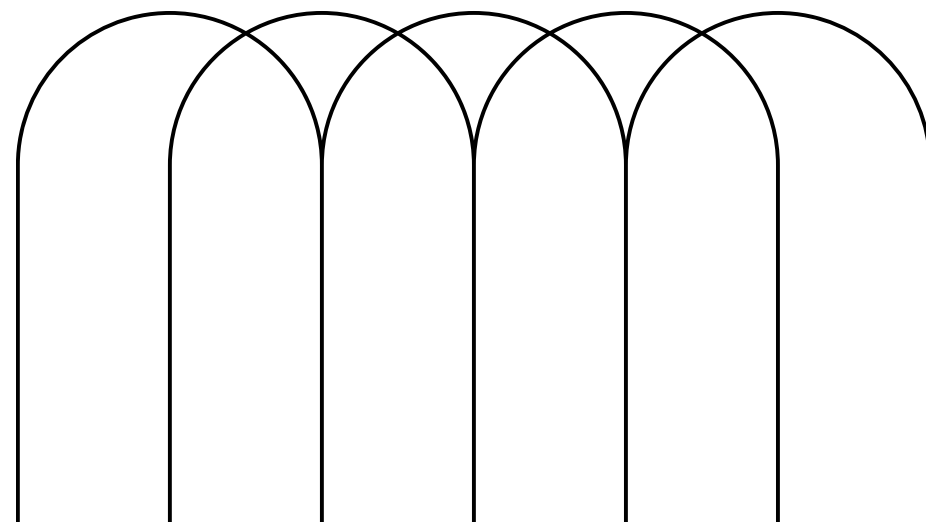


CSS Overflow & Position



CSS Overflow

The overflow property in CSS determines how content is displayed when it overflows the boundaries of its container.

1. visible (Default): This value allows the content to overflow its container without any restrictions.

```
.box { overflow: visible; }
```

2. hidden : This value clips the overflowing content and prevents scrolling or visibility beyond the container.

```
.box { overflow: hidden; }
```

3. scroll : This value adds scrollbars to the container, allowing users to scroll to view the overflowing content.

```
.box { overflow: scroll; }
```

4. auto : This value allows the browser to decide whether to add scrollbars based on the content size.

```
.box { overflow: auto; }
```

5. overflow-x & overflow-y : These properties allow independent control of horizontal and vertical overflow.

```
.box { overflow-y: scroll; overflow-x: hidden; }
```

CSS Position

CSS provides a powerful property called position that allows developers to control how elements are positioned within a web page. Understanding how position works is essential for creating flexible and responsive layouts.

1. static (Default): By default, all HTML elements are positioned static. This means elements are placed according to the normal document flow, without any additional positioning applied.

```
.box { position: static; }
```

2. relative: When an element is given position: relative;, it remains in the normal document flow but can be adjusted using the top, right, bottom, and left properties relative to its original position.

```
.box { position: relative; top: 20px; left: 20px }
```

3. absolute: An element with position: absolute; is removed from the normal document flow and positioned relative to its nearest positioned ancestor (i.e., an ancestor with position: relative, absolute, or fixed). If no such ancestor exists, it is positioned relative to the <html> element.

```
.box { position: absolute; top: 10px; right: 20px }
```

4. fixed: An element with position: fixed; is completely removed from the document flow and positioned relative to the viewport. It does not move when scrolling the page.

```
.box { position: fixed; top: 10px; left: 20px }
```

CSS Position

5. sticky: A sticky positioned element behaves like a relative element until it reaches a defined scroll position, after which it sticks to the viewport.

```
.box { position: sticky; top: 20px; }
```

CSS z-index:

The z-index property in CSS controls the vertical stacking order of elements that overlap. It applies to elements that have a position value of relative, absolute, fixed, or sticky (not static).

```
.box { position: absolute; z-index: 2 }
```

How z-index Works

- Higher values appear on top
- If not set default value is auto
- Works Within Stacking Contexts, a new stacking context is created when an element has a position other than static and a z-index set
- Elements with a negative z-index may appear behind their parent element.



THANK YOU

PHONE NUMBER

(+91) 778 899 2897

WEBSITE

www.indixpert.com