**🧭 Z-Index & Float in CSS**

**🗂️ 1. Z-Index: Control Stack Order**

The `z-index` property controls the vertical stacking of overlapping elements.

**✅ Rule:**

Higher `z-index` value = appears on top

\* Works only on positioned elements\*\* (`relative`, `absolute`, `fixed`, or `sticky`)

**🧪 Example:**

```html

<style>

.box1 {

position: absolute;

z-index: 1;

background: red;

width: 100px; height: 100px;

}

.box2 {

position: absolute;

z-index: 2;

background: blue;

width: 100px; height: 100px;

top: 20px; left: 20px;

}

</style>

<div class="box1"></div>

<div class="box2"></div>

```

HTML

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Z-Index Stacking</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<div class="stack-container">

<div class="box red">z-index: 1</div>

<div class="box green">z-index: 3</div>

<div class="box blue">z-index: 2</div>

</div>

</body>

</html>

CSS

body {

margin: 0;

padding: 2rem;

background: #f0f0f0;

font-family: Arial, sans-serif;

}

.stack-container {

position: relative; /\* Creates stacking context \*/

width: 400px;

height: 300px;

margin: auto;

border: 2px dashed #ccc;

}

.box {

position: absolute; /\* Required for z-index to apply \*/

width: 200px;

height: 150px;

color: white;

font-weight: bold;

font-size: 1.2rem;

padding: 1rem;

box-sizing: border-box;

display: flex;

align-items: center;

justify-content: center;

opacity: 0.9;

}

/\* Red box (bottom layer) \*/

.red {

background: crimson;

top: 20px;

left: 40px;

z-index: 1;

}

/\* Green box (top layer) \*/

.green {

background: seagreen;

top: 60px;

left: 80px;

z-index: 3;

}

/\* Blue box (middle layer) \*/

.blue {

background: royalblue;

top: 100px;

left: 120px;

z-index: 2;

}

**🧠 Notes on z-index & Stacking Context:**

* z-index **only works** on positioned elements (relative, absolute, fixed, or sticky).
* The **higher the z-index**, the closer to the user the element appears.
* A position: relative or similar on the parent (.stack-container) creates a new **stacking context**, isolating stacking inside it.
* **Siblings** inside the same stacking context are layered by z-index.