**🔵 Blue box appears \*\*on top of\*\* red box.**

**🧭 2. Float: Position Elements Horizontally\*\***

The `float` property allows elements (usually images or blocks) to "float" to the \*\*left\*\* or \*\*right\*\*, wrapping other content around them.

**✅ Example:**

```html

<style>

.image {

float: left;

margin-right: 15px;

}

</style>

<img class="image" src="pic.jpg" width="100">

<p>This text wraps around the image.</p>

```

**🔧 Common Values:**

\* `left` – floats to the left

\* `right` – floats to the right

\* `none` – default (no floating)

\* `inherit` – inherits from parent

**🧹 \*\*3. Clear Float to Fix Layout Issues\*\***

After using float, elements below may be pulled up. Use `clear` or clearfix method to fix layout.

**✅ `clear` Example:**

```html

<div style="float: left; width: 100px; height: 100px; background: lightblue;"></div>

<div style="clear: both;">This div appears below the floated one.</div>

```

**✅ Clearfix Method (modern approach):**

```html

<style>

.clearfix::after {

content: "";

display: table;

clear: both;

}

</style>

<div class="clearfix">

<div style="float: left;">Left Float</div>

<div style="float: right;">Right Float</div>

</div>

**✅ Basic Example: Float Left and Right**

**🔹 HTML**

<div class="container">

<div class="box left">Float Left</div>

<div class="box right">Float Right</div>

<p>This text wraps around the floated elements if not cleared.</p>

</div>

🔹 CSS

.container {

width: 90%;

margin: auto;

background: #f9f9f9;

padding: 1rem;

border: 1px solid #ccc;

}

.box {

width: 40%;

padding: 1rem;

background-color: steelblue;

color: white;

font-weight: bold;

}

.left {

float: left;

}

.right {

float: right;

}

**🧠 Summary of float Values:**

| **Value** | **Meaning** |
| --- | --- |
| left | Floats element to the left |
| right | Floats element to the right |
| none | Default — no floating |
| inherit | Inherits from parent |