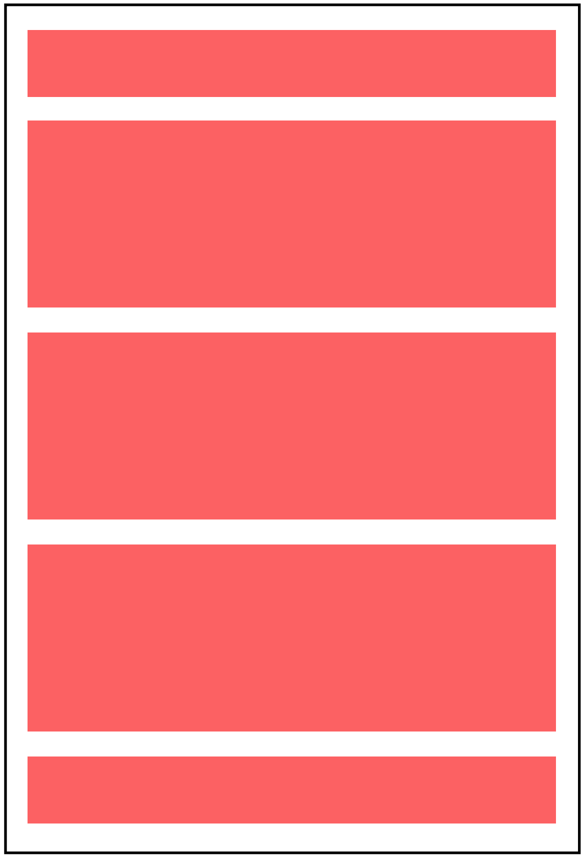
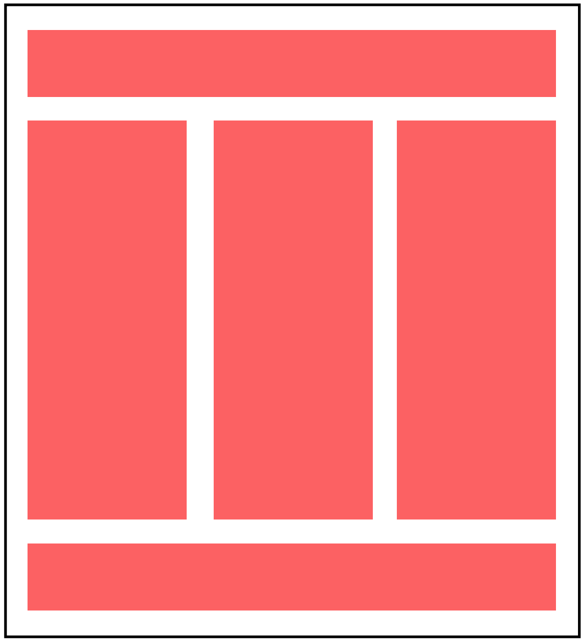
# Floats

An introduction to CSS layouts

Because block elements stack vertically, we've been limited to a single column layout.



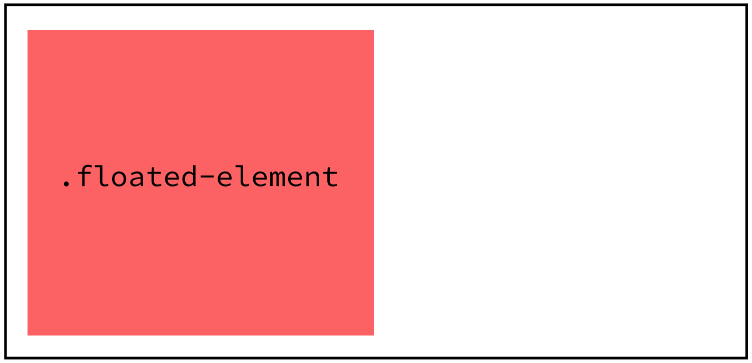
Floats let us put block level elements side by side.



# Float left

style.css

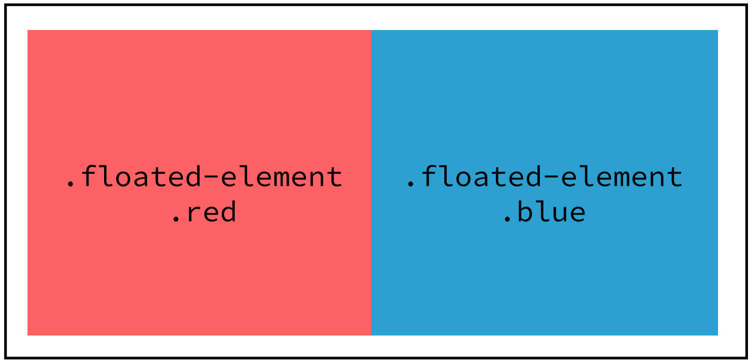
```css
.floated-element {
    width: 50%;
    float: left;
}
```



# Float left

style.css

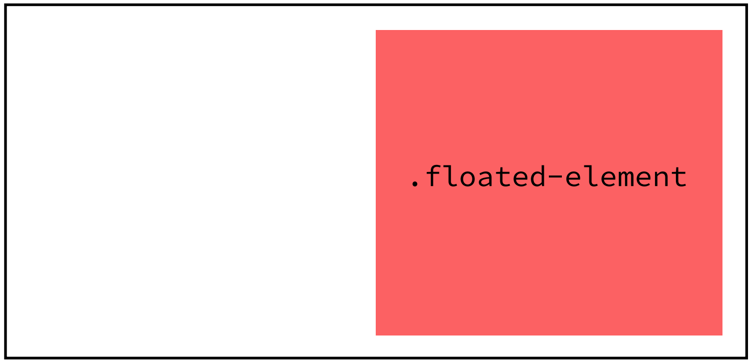
```css
.floated-element {
    width: 50%;
    float: left;
}
.red {
    background-color: pink;
}
.blue {
    background-color: blue;
}
```



# Float right

style.css

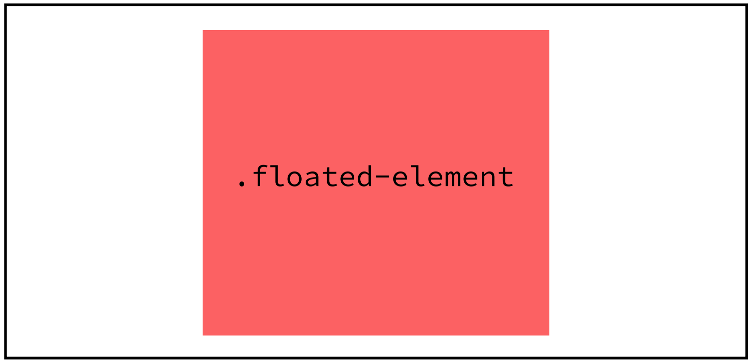
```
.floated-element {
    width: 50%;
    float: right;
}
```



# Center align

style.css

```css
.floated-element {
    width: 50%;
    margin: 0 auto;
}
```



# Clearing floats

“Clearing” a float is when we tell a block to ignore any floats that appear before it.

Instead of flowing around the floated box, a cleared element always appears after any floats.

# Method 1: clear

Apply `clear: both;` to the element that appears AFTER the floated element

index.html

```
<div class=”floated-one”>
    ...
</div>
<div class=”floated-two”>
    ...
</div>
<div class=”clear-this-one”>
    ...
</div>
```

style.css

```
.floated-one, .floated-two {
    float: left;
}

.clear-this-one {
    clear: both;
}
```

# Method 2: overflow hidden

By adding an `overflow: hidden;` declaration to a container div, we’re telling it to recognize the height of any floated elements it contains.

index.html

```
div class=”container”>
    <div class=”floated-one”>
        ...
    </div>
    <div class=”floated-two”>
        ...
    </div>
</div>
<div class=”clear-another”>
    ...
</div>
```

style.css

```
.floated-one, .floated-two {
    float: left;
}

.clear-this-one {
    clear: both;
}
```