



---

# CoderDojo Drogheda – Advanced CSS

---

“The Box Model”

---

---

# CoderDojo Drogheda - Advanced CSS

---

## HTML Crash Course Reminders!

All pages start and end with `<html>`  
`</html>` tags.

The content of a page is placed between  
`<body>` and `</body>` tags.

A div element is like a big box which  
wraps all content within it.

```
<div>My Content Here</div>
```

A bullet point list is created nesting `ul`  
and `li` elements within one another.

```
<ul>
  <li>My first bullet point</li>
  <li>My second bullet point</li>
  <li>My third bullet point</li>
</ul>
```

Elements can also have “attributes” which  
give element additional properties, such  
as an id or class name for selecting by  
CSS. Example:

```
<div id="mydiv">My Dynamic
Content</div>
```

Links to other pages are created using an  
“a” element with an attribute of “href”  
specifying the file to link to:

```
<a href="mypage.html">My Link</a>
```

Some other example HTML elements are:  
head, title  
span  
table, tr, td

## CSS Crash Course Reminders!

CSS is what styles our HTML

Its normal located in a CSS file and

including in our html using

```
<link rel="stylesheet"
type="text/css" href="styles.css"
/>
```

We target elements using CSS selectors:

```
# for element IDs
. for element classes
```

In the style sheet once an element is  
selected rules are then applied within  
brackets:

```
.redboldtext {
  font-weight: bold;
  color: #FF0000;
}
```

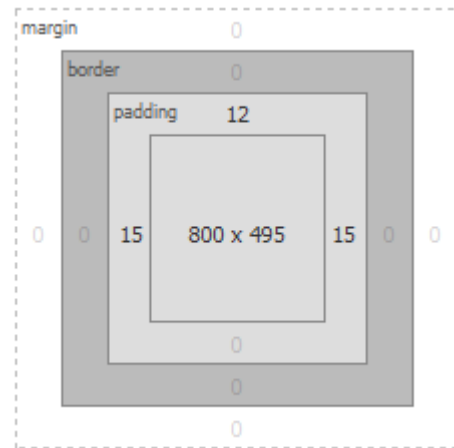
Some other example CSS rules are:

```
font-family: Arial, san-serif;
font-size: 12px;
text-align: center;
background-color: #ff0000;
background-image: url(myimage.jpg);
padding: 4px;
padding-left: 2px;
margin: 4px;
margin-right: 5px;
float: left;
float: right;
width: 100px;
height: 100px;
```

## The Box Model

*This sounds like some mad math formula thing, don't worry, it's not. It just makes it really easy to understanding some styling rules which make your web pages look more awesome!*

Margin and Padding? Why on earth do we have the two?? It seems like they do the same thing! They don't. Some clever fellow back in the day figured out a simple yet genius way to have a box, with a background, some padding inside to keep text away from the walls, a nice border around the box, and a margin to keep the box away from all other elements around it, in one concise CSS styling. We call this, "the box model".



## Example

Say you want to have a text area on your webpage with some text in it:

```
<div id="mybox">My fancy boxed content</div>
```

We want to make it 300px by 300px in size, so we apply some CSS.

```
<div id="mybox">My fancy boxed content</div>
```

```
#mybox {  
  width: 300px;  
  height: 300px;  
}
```

We can't really tell it's a box though, so let's add a border around it.

```
<div id="mybox">My fancy boxed  
content</div>
```

```
#mybox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
}
```

Great! Now we have a box. Let's give it a background colour, and we want to keep the text away from the walls, so let's give the box some padding so that the text sits nicely inside of it.

```
<div id="mybox">My fancy boxed  
content</div>
```

```
#mybox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
  background: #F7CDD1;  
  padding: 10px;  
}
```

So we now should have a box which looks something like the box on the right! (It looks suspiciously like the crash course boxes at the start of this document too..) .

So we can see from our box here, on the outside we have a 1px black border. Then, on the inside we have a padding of 10px around the top, left, right and bottom which keeps the text away from the walls. The inner width of the box is then 300px from the end of the padding on the left, to the start of the padding on the right. We don't yet have any margin...



It's time to get a bit more complicated. Let's have two boxes, side by side, lined up against one another. We'll work with that we have at the moment. One 300px by 300px box, with a 1px black border and 10px padding. Let's add another box to our HTML.

```
<div id="mybox">My fancy boxed  
content</div>  
<div id="secondbox">My other fancy  
boxed content</div>
```

```
#mybox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
  background: #F7CDD1;  
  padding: 10px;  
}
```

And like the last time, we'll add the same box styling to it, giving it a width, height, border, padding and background colour.

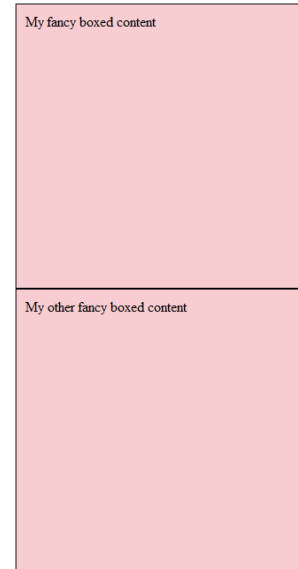
```
<div id="mybox">My fancy boxed  
content</div>  
<div id="secondbox">My other fancy  
boxed content</div>
```

```
#mybox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
  background: #F7CDD1;  
  padding: 10px;  
}  
#secondbox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
  background: #F7CDD1;  
  padding: 10px;  
}
```

We should now have two boxes one stacked on top of the other. Like on the right. The only thing is, this isn't what we wanted. We wanted the two of them side by side, so they're on the same row.

Recall the CSS styling "float". Float allows you to take two block styled elements, like a div, and align them in a row together while still maintaining their block style like attributes. So the box, remains a box, even though other content can me inline after it.

To do this we simply add the CSS rule "float: left" to both of our sets of rules for each box. Now, both of our blocks are side by side! So now we have two boxes, side by side, each with an inner width of 300px, then a 10px border surrounding that, and then a 1px border surrounding each box. But hang on.. how do we put a gap between our boxes?



```
<div id="mybox">My fancy boxed  
content</div>  
<div id="secondbox">My other fancy  
boxed content</div>
```

```
#mybox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
  background: #F7CDD1;  
  padding: 10px;  
  float: left;  
}  
#secondbox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
  background: #F7CDD1;  
  padding: 10px;  
  float: left;  
}
```

Enter: Margin. See, we could just put another element in-between mybox and secondbox, assign it an id, then give it its own styling with float left so that creates the gap between our two boxes, but that's an awful lot of work. Using our box model, we know that outside of the border, we can use "margin". The margin sets an invisible barrier between your element and anything else around it. Think of it as a way of spacing your element away from other elements on the page...

So, let's give mybox a margin to the right of 20px.

```
<div id="mybox">My fancy boxed  
content</div>  
<div id="secondbox">My other fancy  
boxed content</div>
```

```
#mybox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
  background: #F7CDD1;  
  padding: 10px;  
  float: left;  
  margin-right: 20px;  
}  
#secondbox {  
  width: 300px;  
  height: 300px;  
  border: solid 1px #000000;  
  background: #F7CDD1;  
  padding: 10px;  
  float: left;  
}
```

---

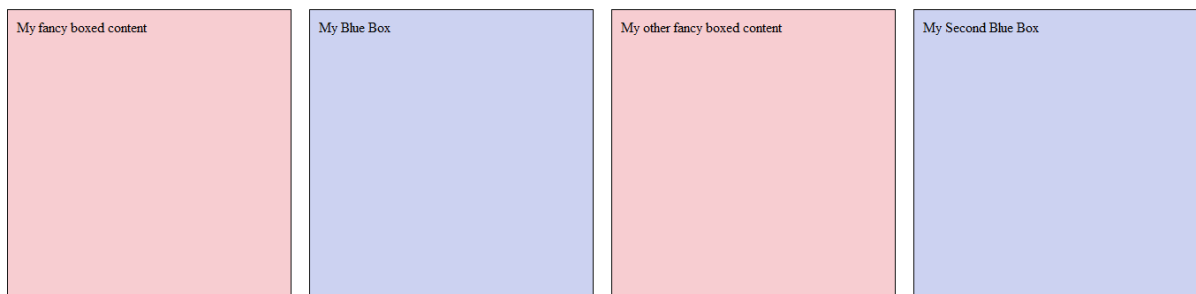
My fancy boxed content

My other fancy boxed content

BOOM! Two side by side boxes, with padding, a margin, a border and a background colour. Notice how when we add the margin to mybox, the background doesn't go with it! That's the difference between "padding" and "margin". Padding sits inside the border. Margin sits outside of the border. The border sits in-between padding and margin.

And that's, the box model.

## Challenge 1



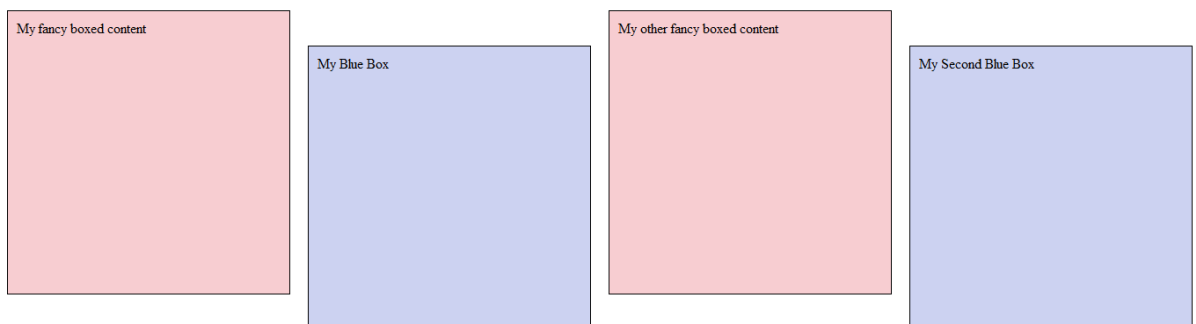
Place 4 200px boxes side by side with alternating Red (#F7CDD1) and Blue (#CCD2F1) background colours like above. There should be a 1px border around the boxes and a 20px margin between each box.

## Challenge 2

Take the boxes you made from Challenge 1, change your css to only have two rules. Two classes called “redbox” and “bluebox”. Apply all the necessary rules to recreate the layout above only using those two rules.

Tip: Instead of setting margin-right on both classes, why not try set margin-left and margin-right only on the blue boxes.

## Challenge 3



Building on challenge 2, recreate the above pattern.

## Challenge 4

Create the Irish flag. Make 3 rectangle boxes, Green (#009A49) White (#FFFFFF) and Orange (#FF7900). You’ll need to make these wider, something like 300px by 200px, then wrap all 3 boxes in another div element and give the outer div a thick 3px black border.

*Tip: You’ll need to set the outer div height and width too.*

