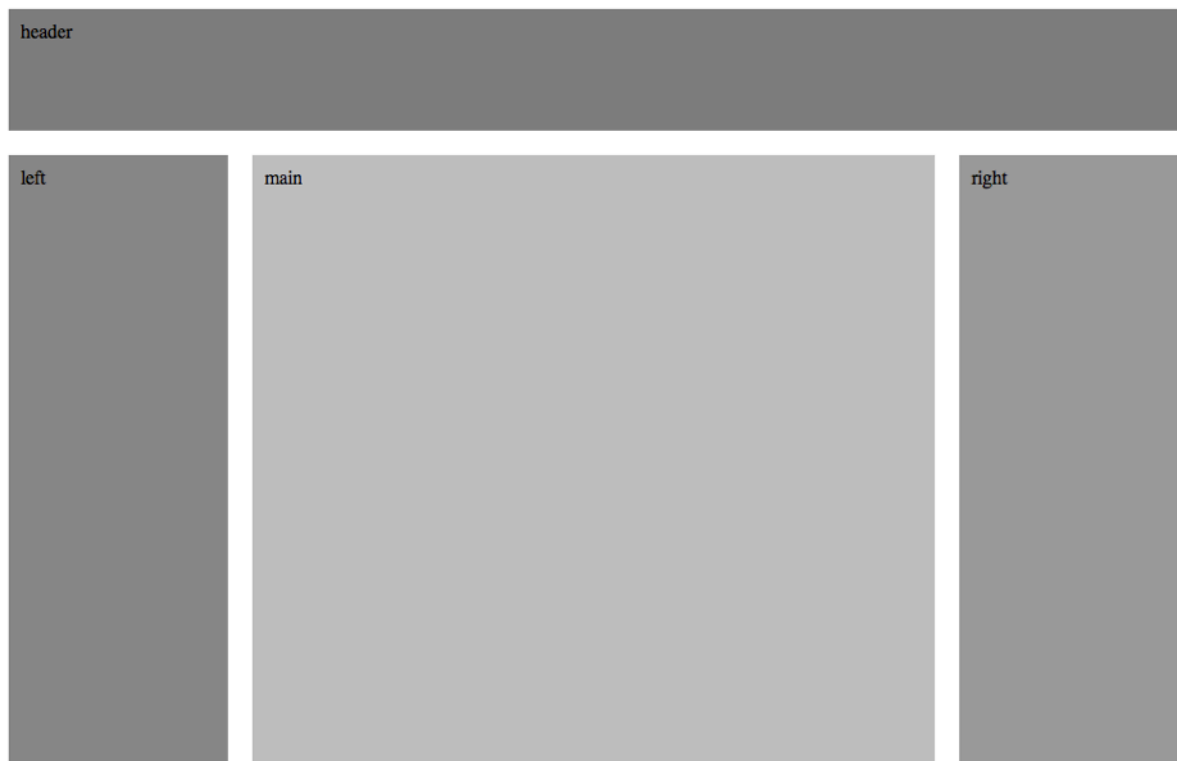


Using the CSS Float Property to Create Multi-Column Layouts

The image below demonstrates the boundaries of the columns in a three column layout. This method of creating layouts is hugely versatile.



An example of a column based layout. Can you think how this might have been achieved?



Where do we begin?

You should have an XHTML template to begin.

In the body of the document, create four divs.

Give each div an id property relevant to the information that would be contained within. So, for example, the header div would be created using the following code:

```
<div id="header">header</div>
```

The name in the id property is what you use to reference each div in the CSS document. The content between the opening and closing tags, in the case the word 'header', is what is displayed inside that section.

Before we are finished with the XHTML page we need to reference the CSS document itself using the following code in the head of the document:

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

(I called my CSS document threecolumn.css, if you call yours something else you will need to change it)

Now let's style these divs using CSS!

Create a CSS document (make sure it ends .css).

Because we have used id's in the XHTML (as opposed to classes) we reference the divs using #s, for example

```
#header {  
    CSS code goes in here  
}
```

By the way, if you were using classes . . .

```
div.header {  
    CSS code goes in here  
}
```

The code we will need . . .

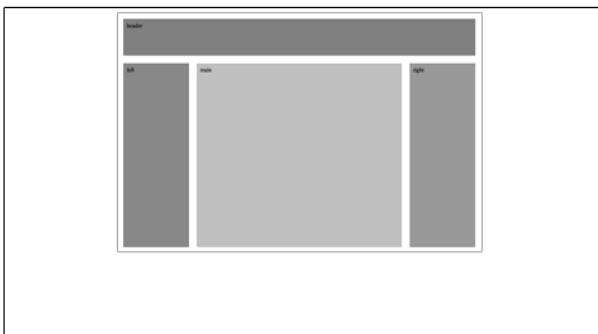
For example

```
#left {  
    padding: 10px;  
    height: 480px;  
    float: left;  
    width: 160px;  
    background-color: #888888;  
}
```

Important!

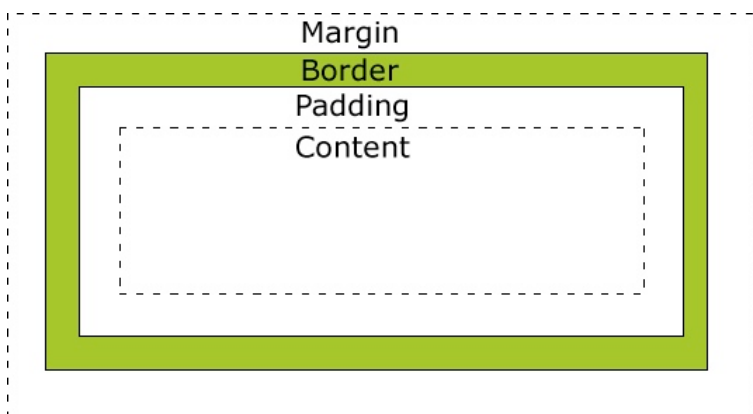
There are a couple of things to remember when floating divs.

- By default, divs are only as large as what is contained within them. This means that we need to specify the height and width in the CSS.
- By default, divs are invisible. To see what's going on with them it is a good idea during the development of the site to set a visible border or to set a solid background colour.
- The header section won't need to be floated as it is going to span across the width of the page.
- When adding padding (see box model) to your divs, the amount of width created by the padding is added on, therefore this needs to be taken into account when calculating how wide you want the elements on your page to be.
- It is a good idea to set margins to the elements of your page, at least during the development phase, to in order that the different elements are clearly distinct from one another.
- You can only float elements left or right, not up or down.
- When floating elements, for example when floating an element left, the element moves as far to the left as it can go. The trouble with this is that more often than not, we want our layout to sit in the middle of the page. What we need is another div to contain the columns and center them. This div is often called a container div or a wrapper div.



The grey line represents the wrapper div. When you are done centering it, you can set the borders to 0 and it'll be invisible.

www.w3schools.com/css/css_float.asp



CSS Box Model

Always keep this model in mind. It is fundamental to good web design.

www.w3schools.com/css/css_boxmodel.asp