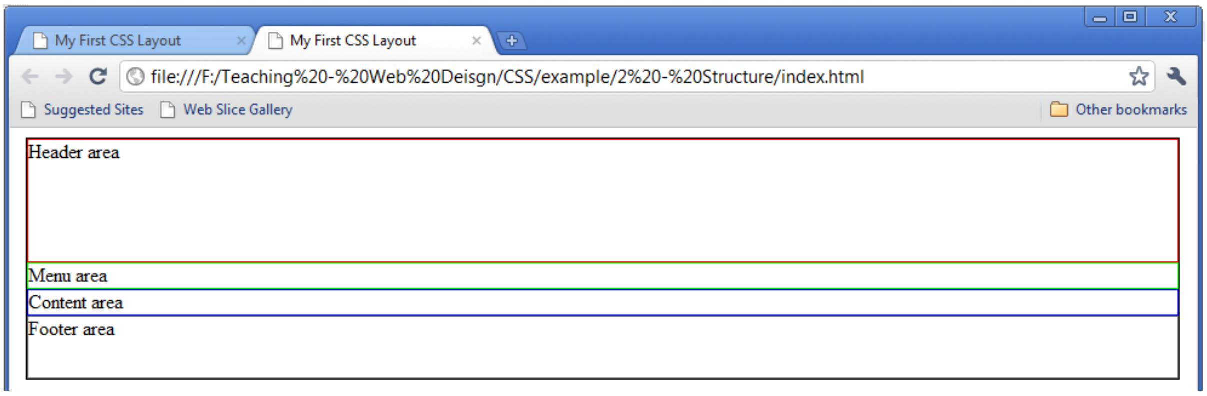
CSS: Behave!

Our bricks are now at the building site, however, at the moment your browser doesn't know where to put them, how much mortar to put around them, or what the bricks are made of in the first place – so it just chucks them all in a pile. Therefore, our website should currently look like this:



Header area

Menu area

Content area

Footer area

The first step towards making your page work EXACTLY how you want it to in every browser, is to take the default 8 pixel border off, which most browsers apply to the <html> and/or <body> tags.

```
html, body {
        margin: 0px;
        padding: 0px;
}
```

A comma in between CSS selectors stands for "or", so here the rule will be applied to <html> or <body>. Because both exist on the page, it will be applied to both.

We now need to float our #menu to the left so that it sits side-by-side with the #content. While so far we have left the <div>s to fill the width of the parent container (#wrapper), we now need the menu to have a defined width. In this case, we'll use 200 pixels. We are also going to need to add some sample content, so we can see what's happening more easily. You can copy and paste the following snippets into your code, but make sure to indent them accordingly, and that you're pasting them in the right places!

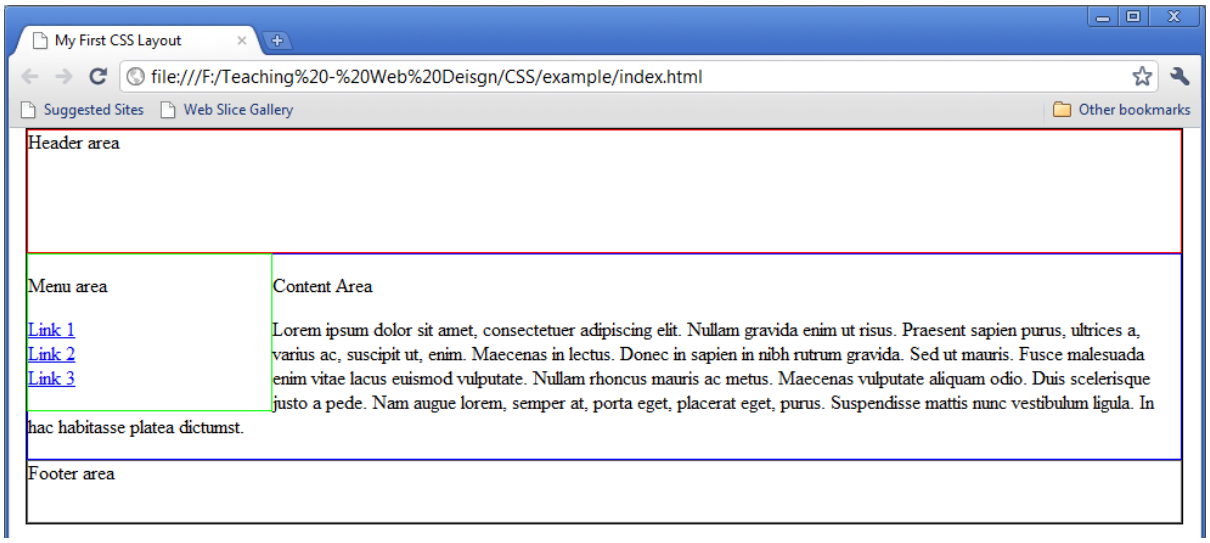
In your HTML File, you will need to update your #menu and #content as follows:

```
<div id="menu">
        <p>
        Menu area
        </p>
        <p>
                <a href="#">Link 1</a><br/>
                <a href="#">Link 2</a><br/>
                <a href="#">Link 3</a>
        </p>
</div>
<div id="content">
        <p>
        Content Area
        </p>
        <p>
        Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nullam gravida enim ut risus.
Praesent sapien purus, ultrices a, varius ac, suscipit ut, enim. Maecenas in lectus. Donec in sapien in
nibh rutrum gravida. Sed ut mauris. Fusce malesuada enim vitae lacus euismod vulputate. Nullam
rhoncus mauris ac metus. Maecenas vulputate aliquam odio.  Duis scelerisque justo a pede. Nam
augue lorem, semper at, porta eget, placerat eget, purus. Suspendisse mattis nunc vestibulum ligula.
In hac habitasse platea dictumst.
        </p>
</div>
```

In your CSS file, you will need to update your #menu as follows:

```
#menu {
        border: solid 1px #00FF00;
        float: left;
        width: 200px;
}
```

We should now see something like this:



Header area

Menu area

Link 1
Link 2
Link 3

Content Area

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nullam gravida enim ut risus. Praesent sapien purus, ultrices a, varius ac, suscipit ut, enim. Maecenas in lectus. Donec in sapien in nibh rutrum gravida. Sed ut mauris. Fusce malesuada enim vitae lacus euismod vulputate. Nullam rhoncus mauris ac metus. Maecenas vulputate aliquam odio. Duis scelerisque justo a pede. Nam augue lorem, semper at, porta eget, placerat eget, purus. Suspendisse mattis nunc vestibulum ligula. In hac habitasse platea dictumst.

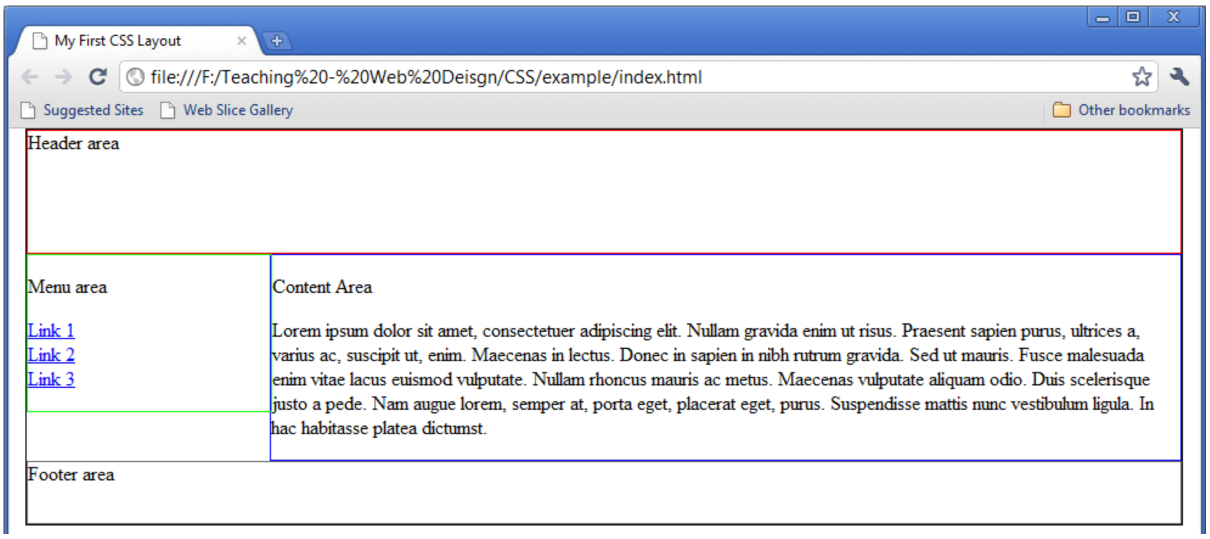
Footer area

Now we don't want our content to "wrap" under our menu – that's just horrible! Let's deal with that now, by adding a margin to the left of our content area, in order to stop it clashing or wrapping.

In your CSS file, update your #content as follows:

```
#content {
            border: solid 1px #0000FF;
           margin-left: 200px;
}
```

Now refresh your page...



Header area

Menu area

Link 1
Link 2
Link 3

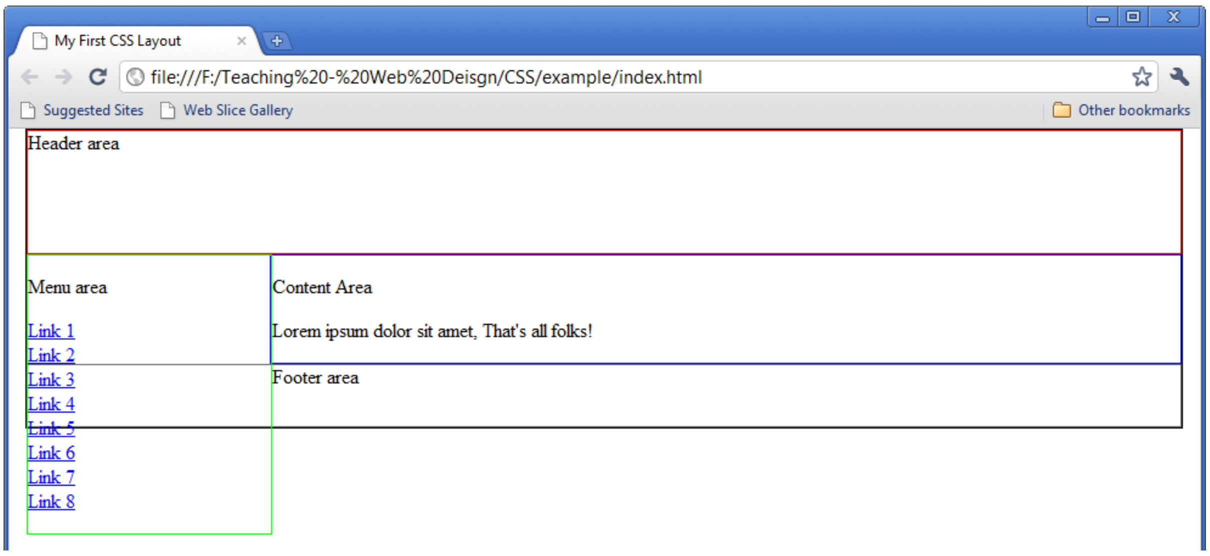
Content Area

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nullam gravida enim ut risus. Praesent sapien purus, ultrices a, varius ac, suscipit ut, enim. Maecenas in lectus. Donec in sapien in nibh rutrum gravida. Sed ut mauris. Fusce malesuada enim vitae lacus euismod vulputate. Nullam rhoncus mauris ac metus. Maecenas vulputate aliquam odio. Duis scelerisque justo a pede. Nam augue lorem, semper at, porta eget, placerat eget, purus. Suspendisse mattis nunc vestibulum ligula. In hac habitasse platea dictumst.

Footer area

Ahhh – that's better!

Now this looks like the end of the story – and it would be for a lot of sites. However, notice the following glitch when you've got a long menu and short page...



Header area

Menu area

Link 1
Link 2
Link 3
Link 4
Link 5
Link 6
Link 7
Link 8

Content Area

Lorem ipsum dolor sit amet, That's all folks!

Footer area

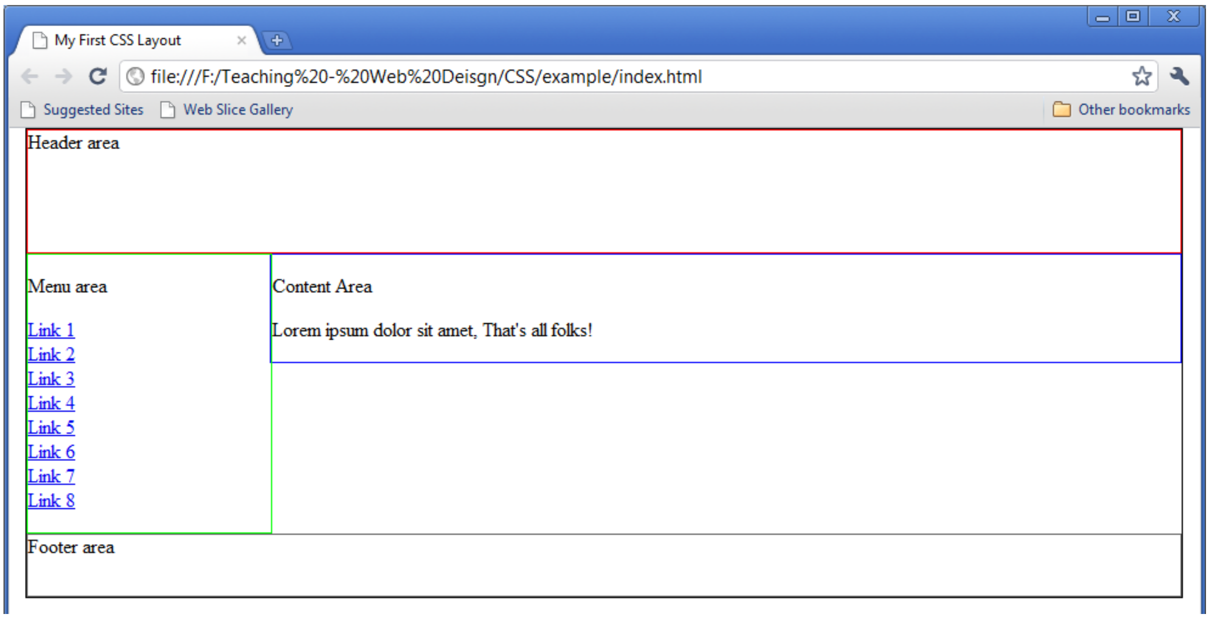
Catastrophe! *(or is it?)*

Not only can this be fixed, but it SHOULD be fixed. Leaving it as it is leaves you open to various other glitches down the line, and should anyone else assisting you, they may find that they create a short page and discover this very problem. The resolution is fortunately quite simple.

In your CSS file, update your #footer to match the following snippet:

```
#footer {
        height: 50px;
        border: solid 1px #666666;
        clear: both;
}
```

When a <div> id'd area has the clear property assigned, if it comes into contact with a floating area it is placed right below where that area ends. You can specify whether it is affected by only left floats or only right floats, in this case we could use either 'left' or 'both'. We'll use the "both" just to be safe, and as it's unlikely we'd want our footer to be overlapped by anything.



Header area

Menu area

Link 1
Link 2
Link 3
Link 4
Link 5
Link 6
Link 7
Link 8

Content Area

Lorem ipsum dolor sit amet, That's all folks!

Footer area

Bingo!

Congratulations – you now have a basic webpage layout!