

How To Get A Fluid Layout, With A Maximum Width

This document is given to you with the understanding that it is original material, written by AJ of [SBI-Help](#), and may not be reproduced without written permission of the author. This is FREE and may be distributed in its entirety if you wish. However, credit must be given to the author. It may NOT be sold.

These directions have been written for those that are using a table based layout for their page design. However, the code and underlying principles can be applied to a full css design.

Why I Have Made This PDF Freely Available

After much searching on the internet, I found there was little or no information to help me achieve the look I wanted on my website template. And if I can't find it, then it's quite likely you can't either! But everyone should be able to get the design they want....and hopefully this report will help you towards that.

Fixed Width Or Expanding Layout?

Currently, you have the choice of either a fixed width or a fluid design. Neither of which is perfect, IMHO. A fixed width will look fine at lower resolutions.....but view a 750px wide web page on a 1680px wide monitor. Especially if it happens to be left aligned. And of course, if it is fixed to a wider width, then those who use 800x600 will have to scroll to view the entire page.

So what about a fluid design.....one that expands to the width of the monitor? Well, that's not perfect either. Once you get past 1280px wide, then the line length becomes uncomfortable for reading, even with a 3 column template (60 characters is about the optimum). Viewed on the same 1680x1050px monitor, the page is possibly better off being fixed at 750px.....you can't win!

If neither of these choices are ideal, what about combining the two? **An expanding page that will only go to a pre-determined width, and then becomes fixed??**

I was originally going to keep this code for my template only. However, after some reflection I felt it was something that should be made available for those that want to try it out.

It works, and it works well....on all OS and mainstream browsers. The code is totally independent of my template, meaning it can be used with any stylesheet. If you want to see the working template, and check out how it looks at different resolutions, then follow [this link](#)

If you decide to implement this way of controlling the width of your page, I would just ask that you consider giving a link to my [SBI-Help](#) page. It's not required of course, but it would be gratefully received!

The argument for using a fixed width layout

- Easy to place everything...you don't need to worry that display results will vary with different monitor resolutions.
- You will set the page so everything looks good at that width....usually optimized for 800x600 resolution.
- Your header will always appear nicely, no problems with the page expanding, but header graphic remaining at its original size.
- Line length will always be at the optimum, no concerns that your page will become difficult to read at longer line lengths.

The argument for using a fluid layout

- You want your page to look nice for ALL viewers, not just the percentage that are using a particular monitor resolution.
- A fluid and expanding page avoids that ugly blank white space on the right, with wider monitors.
- Someone using an 800x600 resolution will have the same optimized viewing experience as someone using 1680x1200.
- As a browser is resized, the content will arrange itself to fit correctly...this won't happen with fixed width.

The Problem

There has always been a problem with trying to achieve this effect. And that is, Internet Explorer (amongst other browsers) isn't standards compliant. So you can get the page displaying correctly in one browser, but incorrectly in another. My way of doing things will avoid this.

What You Can Expect Doing This My Way

My goal was to find a way to get a page to expand for wider monitors, yet not expand so far that the page became **too** wide. But how to achieve this? I searched everywhere I could think of for information, but drew a blank. I could find parts of the puzzle, but not the whole picture. Surely someone, somewhere, had figured out the best way to do this, and documented it?

Well, if they have, the information is well hidden! Maybe it couldn't be done? Or then again, maybe nobody felt it was important enough to spend time working out a solution?

Either way, the end result was that I was on my own. What follows is my solution. You will find it surprisingly easy to implement, and more importantly, you will find it works well with ALL of the mainstream browsers and Operating Systems.

So this is what will happen.

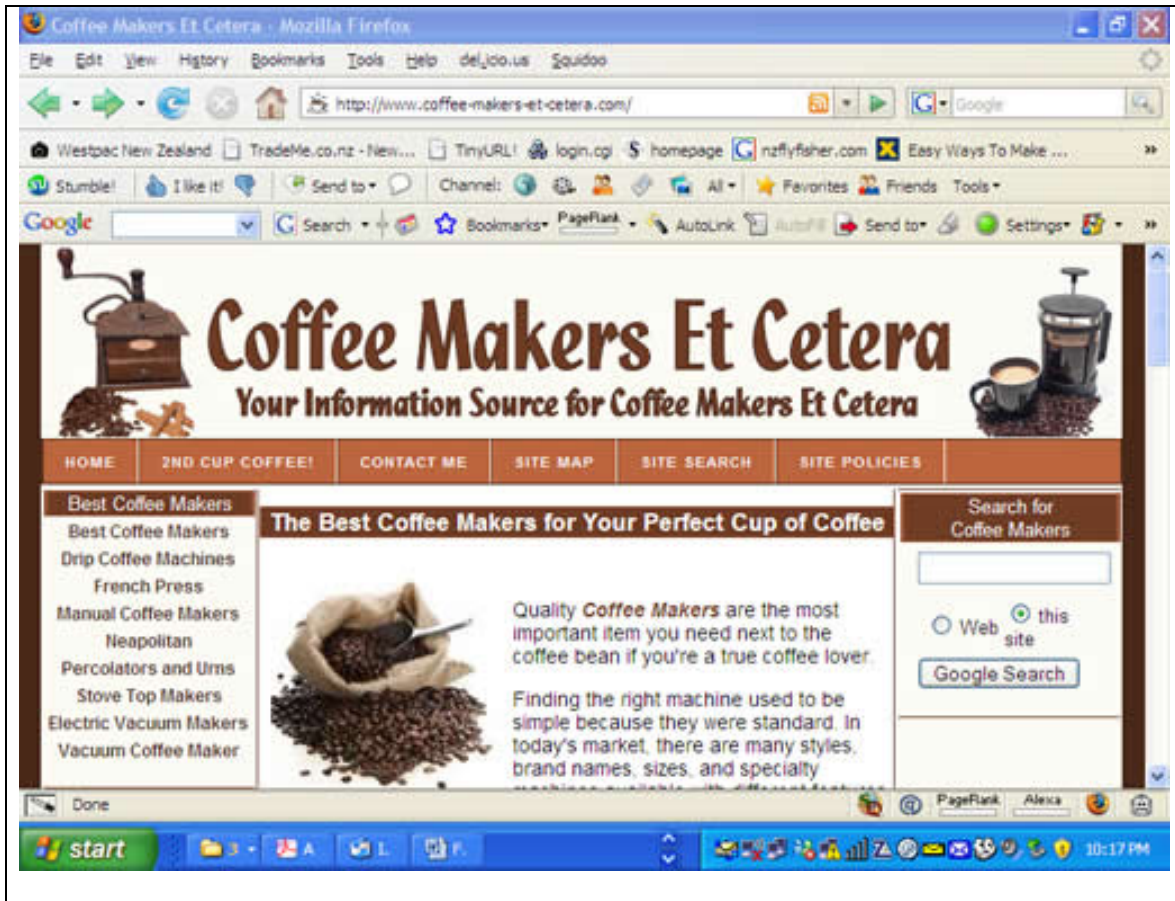
- **You will set a maximum width for your page. Say 1100 pixels**
- **The page will expand to that width, and no further**
- **If the monitor width is wider than 1100px, the page will center, with your choice of border either side**
- **You can ALSO specify a percentage width of the page. So suppose you chose 80%...the page will appear at 80% of the screen, up to your maximum width.**
- **This will happen in ALL mainstream browsers**

An Actual Example

Take a look at the screenshots below, they demonstrate perfectly how a webpage will display using this code. This is an actual web site, not just an example set up for this report. You can visit it at www.coffee-makers-et-cetera.com
The main content table width has been set to 95%. However, the maximum width that the page will expand to is set at 1050 pixels.

This first screenshot is taken at 800x600px resolution (the lowest resolution you really need to consider). You will notice that the page takes up 95% of the screen, with a little colored background each side of the page. So the content area of the page is somewhere around 760px.

Be aware that Internet Explorer deals with padding differently, so the page may be a fraction wider using that browser....enough that you may not see the brown background.



The same page viewed with 1024x768px resolution, appears as below. The page has expanded, but because we haven't reached the cutoff point specified (1050px) it is still taking up 95% of the screen.

So the content area viewed with this resolution is about 950px. (I'm not allowing for the width of the scrollbar here)



Now see what happens when we use a resolution wider than our cutoff point of 1050px! This next screenshot is taken using 1280x1024px.

The content area has expanded only a little wider than the previous shot, to the maximum width of 1050 pixels, and the surrounding body color has filled up the rest of the screen. Pretty neat, huh?



You will also notice that I used MSIE for most of the screenshots. The most difficult browser to get displaying correctly!

And for the last screenshot, I have used 1680x1050px. The only difference with the previous screenshot, is that there is an extra 200px of colored background either side of the page.

A standard 750px fixed width page looks just plain wrong with this resolution, IMO. Especially when it's left aligned.

It also doesn't look good when the page expands to the full width.

Wide screen monitors are starting to become common place nowadays, and as time goes on, there will be more and more of them. So it makes good sense to optimize your page for as many resolutions as possible.....don't restrict yourself to a website optimized for ONLY 800x600, or 1024x768!



Your Stylesheet

You only need to make a few minor adjustments to your existing stylesheet, to get your website looking like the example given. This code has been used on a number of sites now, with NO problems reported, and great results.

This needs to be the first part of your stylesheet. The **max-width** must be there. Also the **text-align:center**; and **margin: auto**; is required to display correctly centered in IE. All other properties are optional. So why is the max-width here, and not down with the table information? More on that later!

```
body {
max-width: 1050px;
background-color:#fff;
font-family: Arial, Verdana, sans-serif;
font-size: 93%;
color: #000;
margin:auto;
padding:0;
text-align:center;
}
```

Next job is to create a class called .main-table

This will set the maximum width for browsers that can't understand the **max-width** property we used earlier.

Just copy the code below into your stylesheet. You can of course change the class name to something more original if you wish. But everything else need to be pasted into the sheet exactly as it is.

```
.main-table {  
width:expression(document.body.clientWidth > 1051? "1050px": "auto" );  
margin-top: 0px;  
margin-right: auto;  
margin-bottom: 0px;  
margin-left: auto;  
}
```

The last part that you need

This part is added to the stylesheet, below the **main-table** class. It sets all the attributes for your main content table, plus any other tables you may add to the page. So just choose the ones you need. You need this, otherwise you will inherit the properties above....things like centering, and auto width.

So paste this code directly below the **.main-table** class

```
table {  
font-family: Arial, Verdana, sans-serif;  
font-size: 93%;  
color: #000;  
padding-top: 5px;  
padding-right: 5px;  
padding-bottom: 5px;  
padding-left: 5px;  
}
```

You can of course change the fonts, sizing, margins etc to suit yourself, but this will give you something to go with in the meantime.

One Addition To Your HTML Page

As you now have a new class in your stylesheet, you need to call it onto your page. So directly after the **<body>** tag, add this to the page.

<div class="main-table" align="center"> and then add the closing div **</div>** just before the **</body>**

That is the only change you need to make to the actual page. Note that you can STILL use a width property in the main table tag. EG: **width="90%"**

What Will Happen To Your Page Now

Tip!

You can change the max-width, and the width in the expression to whatever you want. But they must be the same. Change one to 1200px, then the other needs to be 1200px as well! Otherwise you will have different display results, depending on the browser being used.

If you use the code provided, with no alterations, this is what you can expect.

Your page will appear full screen, (assuming you haven't changed the table width to 90% or something). It will keep expanding until it reaches the 1050 pixel cut off point.

When it gets to 1050px width, it will stop expanding, and the page will center, with a surround of your choice either side of the page (in this case, it's a white background...#fff)

You will get this happening in ALL browsers. (the exception is IE5 which left aligns the page, although it still applied the width attribute.)

This is something that has previously been difficult to achieve. If you find a mainstream browser that this DOESN'T work with, please let me know. Then I can work on a fix for it. (it won't work with IE4 or earlier, but the percentage of people using that is tiny - I don't intend to spend time on obsolete browsers such as this).

You maybe concerned that if you set the main table width to say 80%, that it will cause problems. Don't worry, it will work just fine! For full screen display remove the width="100%" property from your table tag on the pages, if you have this attribute.....it will default to 100% width. Only use the width attribute for less than 100% width.

How And Why It Works

Firstly, look at the **max-width** that I have placed in the body section of the stylesheet. Why have I placed it there, rather than down with the rest of the table information?

It could actually go down there, but it then becomes a lot more complex with centering, and won't work well with the other code placed there. (*some browsers give all sorts of problems!*)

Here's what happens.

Down in the **.main-table** class, I have set the left and right margins to auto. That means the table will expand to full width of the body.

Because the body and the table are now identical widths, by applying the max-width to the body rather than the table, I avoid all the un-necessary complications that go with trying it that way.

So really, it's the BODY I have kept to the max-width, not the content table at all. The background color is behind the main content table and is full width. So at any time that the page isn't at full screen, it will show on either side.

It means there is also no conflict when I want to change the table width to say 80% on the page.

In other words, suppose I did it that way, and tried to set the table width to 80%, but had the max-width set to 1000? If I were viewing on a wide-screen monitor 1600px wide, I *could* end up with 80% of 1600, instead of 80% of 1000px on some browsers.

If that makes sense, then great! If not, then don't worry, it will all work just the way it's meant to, whether you know the mechanics of it or not.

The Expression Part Of The Code

This is for non compliant browsers such as IE 5,6, and 7. These browsers don't understand max-width.

It's a javascript that will give the same results as a **max-width**.. Take a look at it.

width:expression(document.body.clientWidth > 1051? "1050px": "auto");

basically, it is saying, is the page size wider than 1051 pixels? If so, then make it 1050px wide.

If you are taking notice, you will see I have actually made the two widths different. One is **1051**, and the other **1050**.

That is to fix a bug in IE6 that loops the code if the browser is resized, when both are set the same. This can result in a frozen screen. It only happened when the browser is being resized, and hits that critical point.

So with Internet Explorer there is a very slight jump from 1051 to 1050 when resizing, but it's barely noticable. The viewer would have to be using IE, have a screen resolution greater than the width you have specified, and must also be in the process of resizing the browser, before they see the jump. And a jump from 1051 to 1050 pixels is so small you won't even see it, unless you are expecting it.

The results are very good.....you get the benefits of a fixed width, and the benefits of a fluid design. With none of the drawbacks of either.

Using A Full CSS Layout?

This should work equally well with this type of design. The **max-width** goes in the body section of your stylesheet, the same way as for the table layout.

And where the **.main-table** class is used, create a div that will wrap around your main content container. The only difference is, that you are affecting your div "container" rather than a main content table.

So you would have on your page something like `<div class="this new class with the expression in it" align='center">` with that class containing the expression, and the margins. Exactly the same way as done with a table layout.

You would put this directly after the `<body>` tag, and the closing div directly before the `</body>` tag.

(note: you can actually put the class BEFORE the <body> tag, and the closing </div> after the </body>...it will work exactly the same way)

One note

Having a javascript expression in a stylesheet means that it WONT pass validation. However, you can blame Microsoft for that, as they seem unable to design a browser that recognizes css2!

Hope you get some benefit from this, and please let me know how it goes for you.

To try it out, set the max-width and the expression to a size less than your browser resolution. If you are using 1024x768, try it at 900px. Then remove the width percentage from the page table tag, if you have one.

This means the table will expand to full screen width. If you see that the page only goes to 900px wide, then everything is working as it should. In other words, the page should expand to the width you set of 900px, and no further. Check in both Firefox and IE. If things look OK with those 2, then it will likely appear good in other browsers as well. I have checked all the main browsers, with no problems.

You can always take a look at your page using <http://www.browsershots.org>

Once you are happy it's working correctly, my recommendation is for the width to be set somewhere around 1050-1100px.

That is because for a 3 column layout, with a standard size font, you won't exceed a comfortable line length.

Have fun with this. Cheers, AJ ☺