

# More Than One Way to Skin a Web

By Isaac Castillo

<http://www.tri-omni.com>

icemancast (at) gmail dot com



It's been a while since my last post and I really hope this makes up for that :).

If this tutorial was helpful please [Digg This!](#) Also if you want to link to it from you website simply copy the code below

```
<a href="http://www.tri-omni.com/tutorials/more-than-one-way-to-skin-a-website/">More than one way to skin a web</a>
```

In in a past post I mentioned that I would post the way I go about taking a design from Photoshop and slicing it up into a website CSS. I have created a simple design to go through the process that I go through when taking a design from Photoshop to CSS. The way I do this process is not the only way to go about skinning a website. It is most efficient that works for me. I will try to remember all the sources that I have gathered information from and post them here. Of course I created a source zip file for you to download for free.

 [Download working files](#)

 [Download PDF of this tutorial](#)

## 1.) Slicing the design

First I take a look at the design to figure how I am going to slice it up. Remember there isn't really a exact way to do this other than just making sure you are keeping the end user in mind. When I was learning CSS about 1 year ago I really couldn't get my head wrapped around the subject coming from a table based design background. I found [Boag World](#) and began posting for help. [Go to the post](#). The links within the post may not work anymore but there is some good advice that I received from PaulH and Rich Quick ([Web Design Cornwall](#)). Richard Quick explained how websites are like Ogars and Ogars are like onions just kidding he explained how websites are like onions read his reply to my post below.

me: how should i think of design. top to bottom left to right.

richquick: Think of it like a russian doll or an onion skin.

Start from the outside and peel away the layers.

The first thing to sort out is the >body< tag - make sure you set the margin to zero and set the background you want.

Then, write a div, with and ID of "wrapper" which will hold the whole of the design you did in Photoshop.

Then, starting at the top, create the main sections of the page using divs. Eg, for a typical page (2-column layout with horizontal nav):

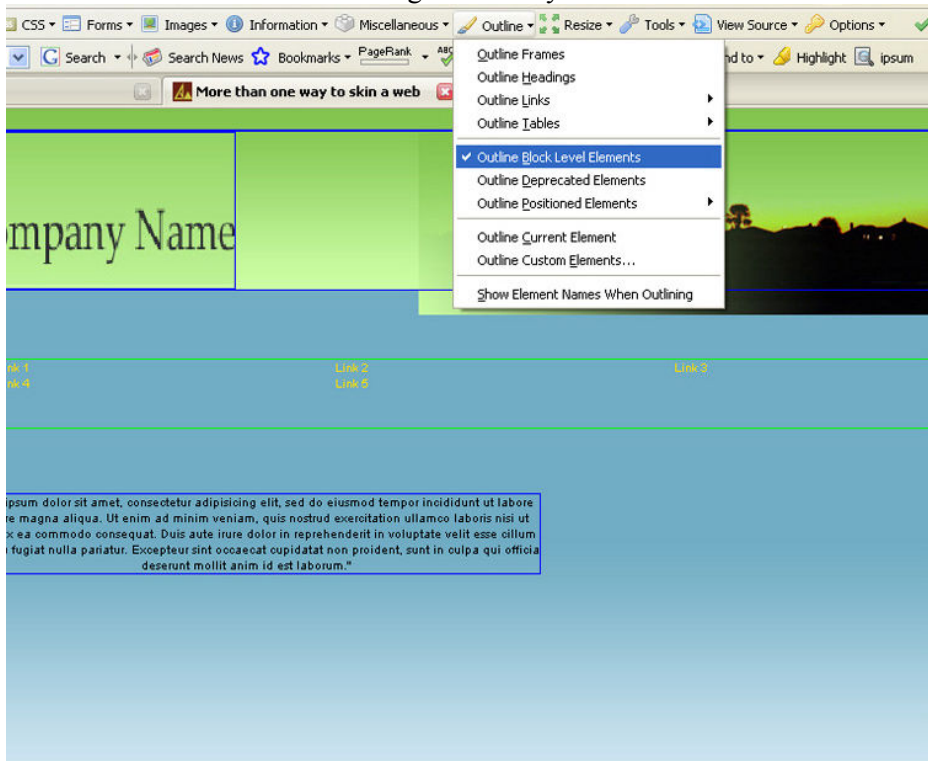
1. header
2. main nav
3. main content
4. sidebar
5. footer

After some research and some CSS read like [CSS Mastery](#). I have come about my own way of slicing the images. I first tend to study the design background images keeping the above list in mind. Can this background be repeat-x or repeat-y, if not then I will have to save the whole thing as an image background or can it be represented as a color.

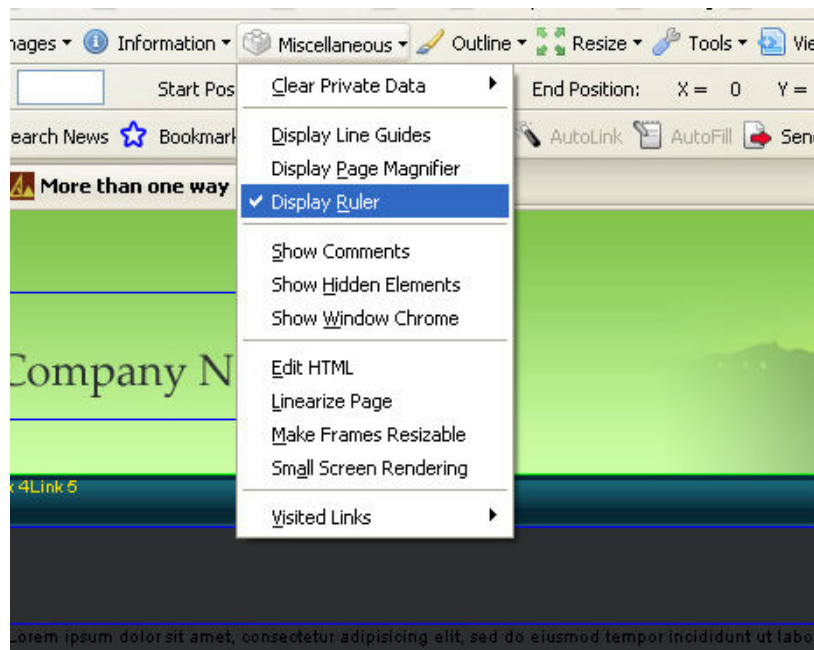


## 2.) CSS & XHTML Templates

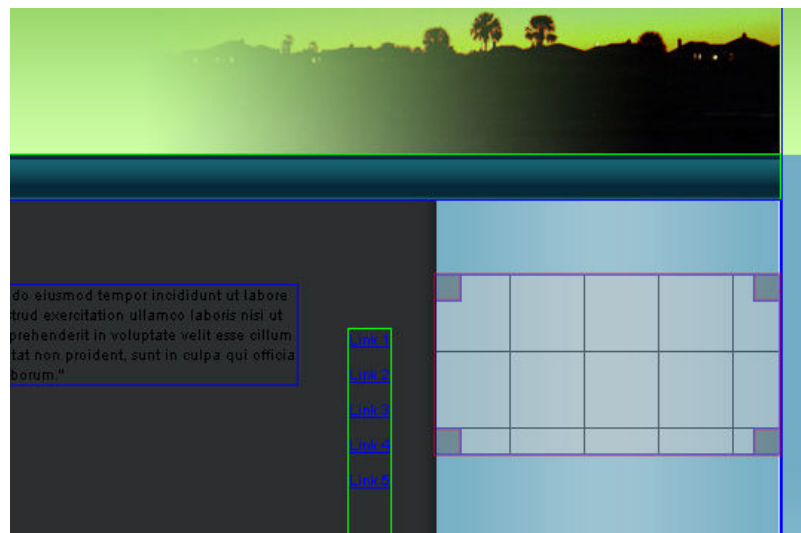
First let me start this section with saying if you are not using [Firefox](#) and you are design websites you are totally missing out on what it has to offer. One of the add-ons that use all the time is the [Web Developer](#). Whenever I am starting the shell of my CSS design I turn on the outline block level elements so I am able to view each location of the div tags within my browser.



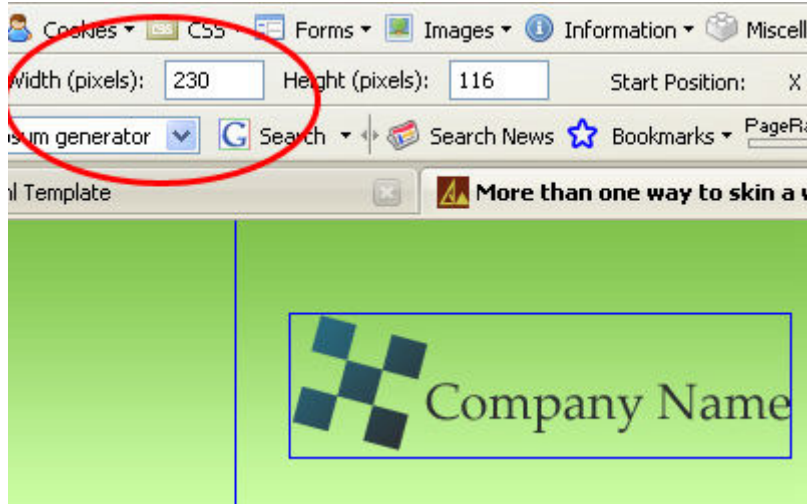
I also tend to have Firefox open along side IE7 and IE6. You can install a version of [Multiple IE](#) or to check how your design looks in several browsers you can use this awesome browser app called [Browser Shots](#). I have template html and CSS files I use for all the websites that I create, meaning I have the infrastructure in place and copy this code over to make things move faster. You can view my [template files here](#), these will should always remain here if you want to link to them or use them. After these files are applied and my design is sliced up I get to the nitty gritty of messing around with the CSS code. The structure of my CSS template was taken from the [CSS Mastery](#) book, once again a great book to start with if you are trying to learn CSS (remember I didn't know much CSS about a year ago). While I am messing around with the CSS code other tools from the Web Developer add-on come into to play. Another main tool I use is display ruler.



The measuring tool in action.



The actual measurement.

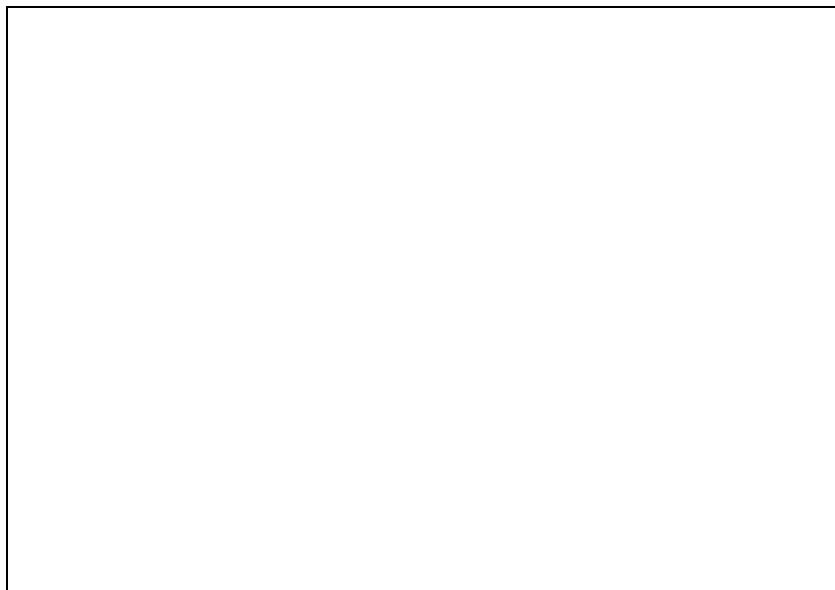


### 3.) CSS Work

After my template files are applied I launch my browser and begin making changes to get the results I want. If you understand how the [box model](#) works you are well on your way to understanding CSS. There really is not much I do to the template files other than messing around with dimensions, margin and padding. There are times that I need to add line or two if I need to add a new div but this depends on the project that I am working with. The hover images are handled in the a:hover CSS inside the mainnav tag. I learned to use CSS to preload images so the swap would happen instantly [here is the post](#) that I learned it from.

### 4.) Cross Browser Checking

You will find that your design looks great in Firefox, okay in IE7 (it is getting better) and whoa what happened in IE6! Another great page to look at is [browser statistics](#) page which lists the most recent browser usage.



Firefox Render



See the difference. Right away you may think oh that is easy just add a conditional statement and I very well can but I tend to take it a step further and try to fix the problem in my regular css file first. If I can not fix the error in my regular css file then I create a conditional statement. How do debug a CSS file? The same way you debug code. Remove a line see what happens, add some code and see what happens and set a width and see what happens. It literally is saving the file over and over again and refreshing the browser over and over (control + f5 is a hard refresh) till results start looking pleasant. In this case I gave a width to the "#mainnav li" item. I could have used a conditional statement but try to avoid that if I can. Below is an example of the conditional statement I use for IE6.

```
<!--[if lt IE 7]>  
<link href="http://www.tri-omni.com/wp-content/themes/esbic/ie6_style.css"  
rel="stylesheet" type="text/css" media="screen" />  
<![endif]-->
```

## Conclusion

You will notice that the footers look different in Firefox and in IE7 of course first thing I would do before adding a conditional would be to figure out why in my original CSS document. Sure the first would be easier but the strive and be a better professional I prefer the latter. I hope that this information was helpful to you and please do share this post by [Digging It!](#). This is information that I have gathered over the course of a year from learning CSS. Also comment here to let me know if this was helpful.