The Guide for Standards-Based Web Design

by:
John G. Hendron
biberfan@mac.com

© 2004 by John G. Hendron. All Rights Reserved.

The Guide
for Standards-Based Web Design
with Macromedia Dreamweaver and Adobe Photoshop

by John G. Hendron,
curator of http://hendron.is-a-geek.net/ and http://www.biberfan.com/
Webmaster, Goochland County Public Schools (http://www.glnd.k12.va.us/)

Introduction

Web Standards are a group of rules folks have agreed upon as the "way" to markup pages for delivery through the Web. The organization that codifies these rules is the World Wide Web Consortium (W3C), and its director is Tim Berners-Lee, the creator of the Web. The consortium is made up of a large group of individuals, including those representing major technology corporations, such as Microsoft, Apple, Adobe, and Macromedia (the full list is online at: http://www.w3c.org/Consortium/Member/List). Among the two most important standards that we read about today are the markup languages XHTML and CSS.

XHTML is the successor to HTML, and the letters stand for *eXtensible HyperText Markup Language*. With attention cornered on the novelty of "X" themed gear, such as X-Box. X-Men and Mac OS X, the "X" certainly adds interest to boring old HTML. The syntax of XHTML is slightly different from its little brother, and follows syntax rules belonging to XML, the *eXtensible Markup Language*. Knowing simply that XML is a powerful markup language on its own, and that XHTML is a subset of this, promises that well-coded XHTML will last the test of time.

One of the things the W3C did was to remove presentational markup from HTML, when codifying XHTML. This eliminates tags such as FONT, BGCOLOR, and discourages the use of spacer GIFs, * * spaces, and the like. Instead, the W3C recommends the use of *Cascading Style Sheets* (CSS) to change the presentational appearance of content on the web. Because CSS is separate from the structural format of your documents, the content can be presented on a variety of media, even on devices that do not understand CSS. When XHTML is combined with CSS, you have the opportunity to share your content with as wide an audience as possible, using browsing devices other than a PC. CSS also allows you to present your material in different guises, to offer, for instance, different backgrounds, color schemes, and font sizes.

Using web-standards is in large part also helpful for creating accessible web sites. Many folks who browse the web have disabilities, which range from a lack of a full range of motor skills (for manipulating the keyboard or mouse), to vision, and hearing. Pages that follow web-standards can be easily modified, if need be, to meet government-sanctioned requirements, along with those presented by the W3C, for accessibility (http://www.section508.gov/).

Tools

There are a variety of tools you can use to generate web sites, and many of them will allow you to effortlessly do so using Web Standards. Needless to say, no product will automatically do this for you; it takes knowledge of the standards to make sure the tools follow your desires for standards-compliance. This guide will cover two of the most popular tools available: Adobe Photoshop for graphics, and Macromedia Dreamweaver, for XHTML and CSS. The versions covered for this guide are Photoshop 7 and Dreamweaver MX. Screen-shots are from the Macintosh (OS X) versions of these cross-platform applications.

Simple XHTML Rules

The differences between HTML 4.0 Strict, another web standard, and XHTML 1.0, are few. The following are a few tips to produce good XHTML markup. There will be exceptions to these tips and an extensive text on XHTML will be your best guide. I highly recommend *HTML & XHTML, the Complete Reference*, by Thomas A. Powell, and published by Osborne Complete Reference Series.

- All markup is in lowercase
- All tags close, even single tags such as `<br />`
- Presentational tags and attributes are out (`bgcolor=, vlink=,` <FONT>)
- The document opens with a `<!DOCTYPE>` tag (which incidentally, breaks rule #1)
- A tag declaring character encoding is included (telling which set of characters you're using)
- All attributes use quotation marks around values (title="Visit our Store")
- You use the `alt=` attribute with the `<img>` tag, to give context for the image in lieu of displaying the image

If you put Dreamweaver into XHTML mode, it does a fair job at providing help with many of these rules. You can force it to include a <!DOCTYPE>, add quotes around attributes, and make it bug you for ALTs in your IMG tags. Your ultimate help, however, will come from the W3C validator, a server-based application that checks your web page for standards-based compliance (http://validator.w3c.org/).

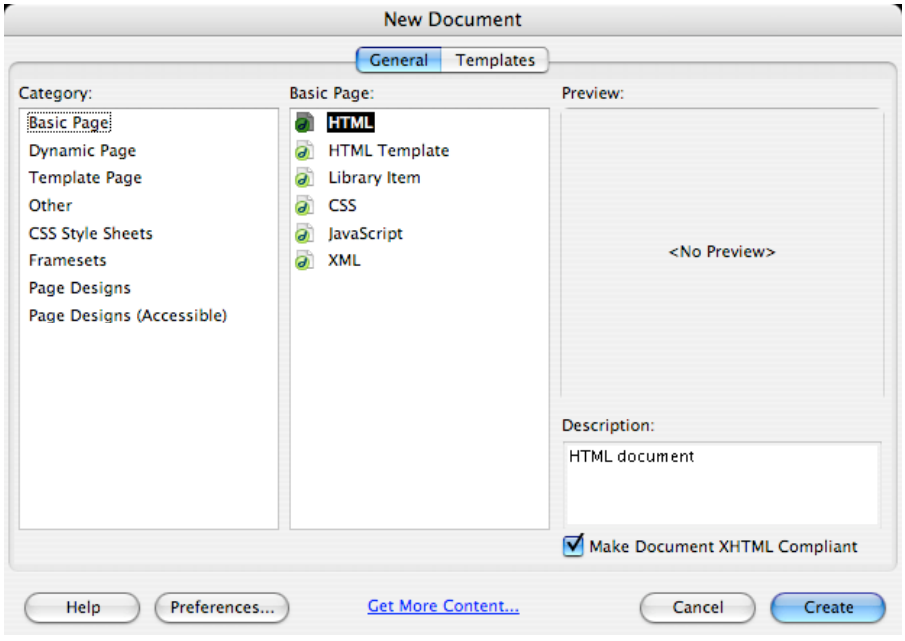
The following code, borrowed from the Goochland County Public Schools website, demonstrates what the <HEAD> of an XHTML document might look like:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
        "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
      <head>
      <title>Goochland County Public Schools Homepage</title>
```

Notice in the <!DOCTYPE> I've indicated the "flavor" of XHTML I'm writing in: XHTML 1.0 transitional. This means I can break a few rules, and still get-by with using XHTML. One drawback in the Goochland site was the use of a <TABLE> for presentation. A table-less version had first been created, relying upon CSS, but failed to work in every browser. Using a hybrid solution aided in the support we could offer to families using a wide-variety of browsers. More <!DOCTYPE> tags can be found online at: http://www.webstandards.org/learn/templates/index.html. Compare the features supported by different browsers at WebMonkey: http://hotwired.lycos.com/webmonkey/reference/browser_chart/.

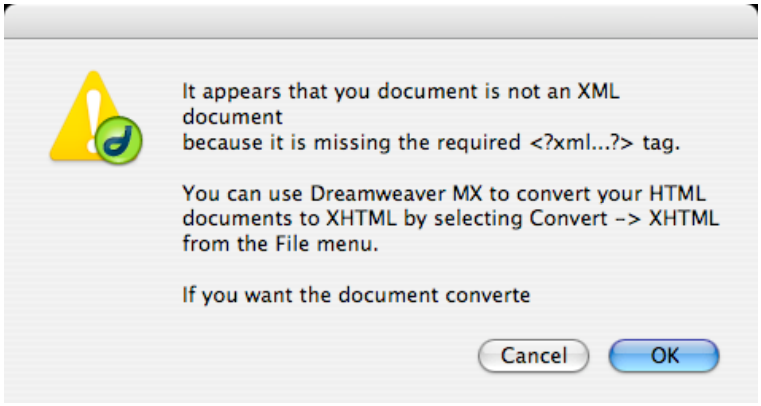
If you're familiar with HTML, you might find this article a good place to start, which compares XHTML and HTML, and what they have in common: http://www.webstandards.org/learn/reference/common_ideas.html

Into Dreamweaver



From Dreamweaver, we have the New Document… dialog box. While you can build your own templates, with any such starter markup you choose, notice here we have chosen an HTML page, and have checked the "Make Document XHTML compliant" option.

You can also have Dreamweaver try to convert documents you already have created in HTML, to XHTML:



The code I start with, after choosing an XHTML Strict definition, is:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
        "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"
/>
</head>

<body>

</body>
</html>
```

My recommendation at this point is to remove the first line, declaring which version of XML is used. Why? Some browsers cough on it, and it may give you problems. It's in the specification for XHTML, but not required for version 1.0.

Notice the <META> tag closes by using a space, then `/>`. Other tags this affects are `<br />`, and `<hr />`.

One thing you'll notice when looking at pages coded in XHTML (and I do mean here, looking at the code), is that they seem very clean, and perhaps even simple. This is because most pages have a straightforward structure: headings, paragraphs, and some with sections (<DIV>). All the code for presentational markup is either embedded in the <HEAD> section using CSS, or neatly tucked-away in an external style sheet. While many designers think of creating pages with a visual emphasis, it is always refreshing to think about your page in terms of text flow, and the focus upon the quality of writing. Simple, clean markup then usually follows. For tips on writing for the web, I highly recommend *Hot Text: Web Writing that Works* by Jonathan and Lisa Price, and published by New Riders.

CSS Magic

Cascading Style Sheet language books again might be a good, all-around reference if you want to learn all CSS can do, and none are better than those authored by Eric Meyer (http://www.meyerweb.com/). For some reason, there are a lot of sites dedicated to CSS information, and you might be just as happy consulting the Web, plus it's free.

CSS has three main ways to make style definitions:

- by tag
- by class
- by ID

CSS also includes three different types of style sheets:

- inline
- internal
- external

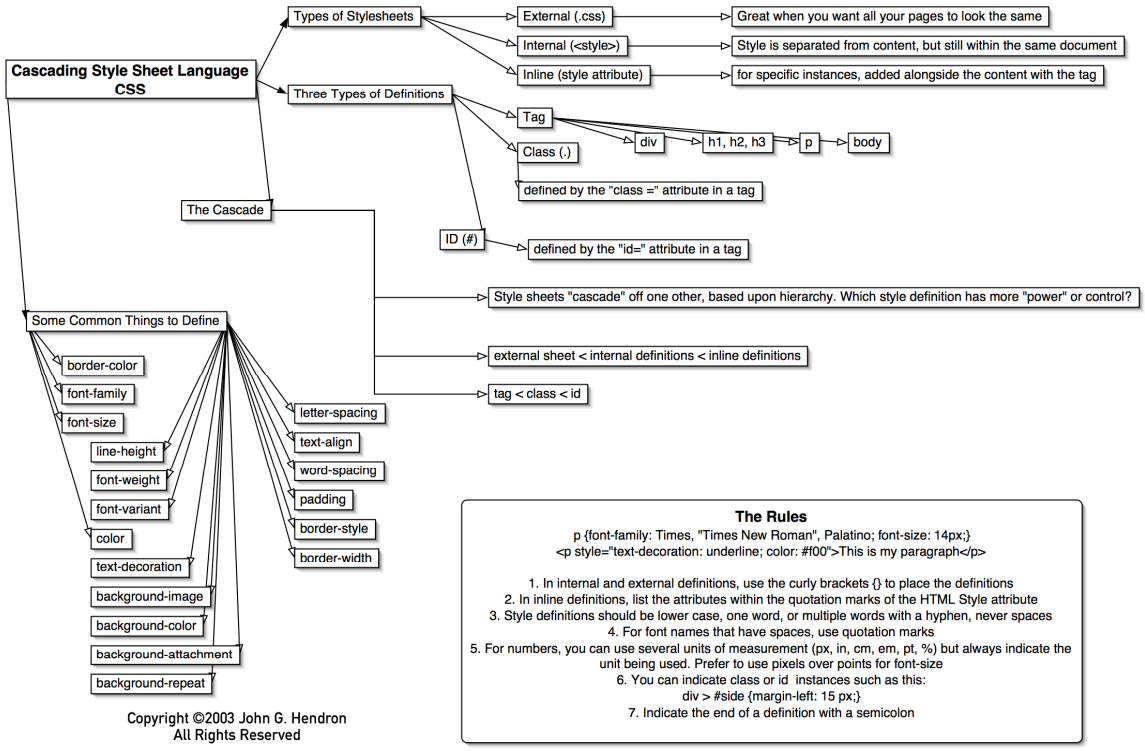
The diagram following this section tries to sum-up a lot of information about CSS in a small space. If you want to know CSS inside and out, this diagram will probably appeal to you—or use it as a reference. The "cascade" in CSS means that the rules you apply to how something looks (say a background, or the font size of your paragraphs) can "cascade" off one another if they are defined differently. For instance, let's take a look at the following definitions:

```
p {font-size: 15px; color: #000; }
p.important {font-size: 20px;}
p#first {font-size: 15px; color: #333;}
```

The first definition says: Size all paragraphs with a font with an x-height of 15 pixels, and make the color of the text black. The second says: for paragraphs of the class "important", change the font-size to be 20 pixels, and keep the black color. And, the third: make the paragraph with the ID

"first" dark gray, and we'll keep the 15 pixel size. My question to you is: What would a paragraph, defined as such, look like?

```
<p class="important" id="first">This is my first paragraph.</p>
```



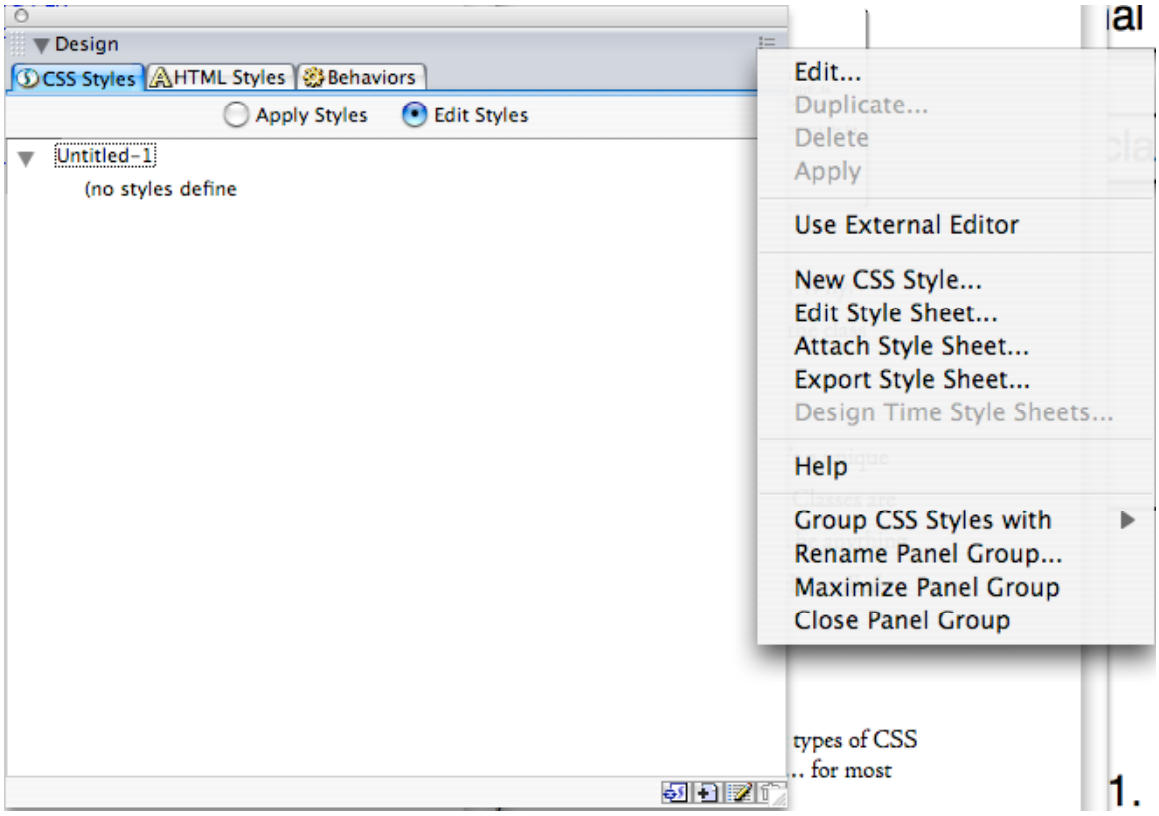
According to the rules, it would have dark gray text and a 15-point x-height in size. Why? According the diagram, a rule that's defined with an ID overrides class, and in this case, the class definition gets overruled!

If you're new to CSS, keep in mind: you only have one "ID", typically, in your page. It's a unique object, and you can ID just about any tag… an image, an <H2>, or whatever you like. Classes are the same way, but are not unique. And the names you assign both classes and IDs can be anything you like, save for reserved words. You cannot use the same identifiers, either (you wouldn't create a class called "X2" and an ID called "X2."). Don't try to confuse your web-standards browser.

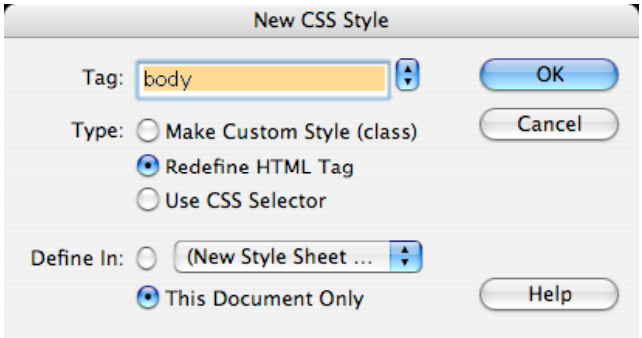
Dreamweaver and CSS

Thankfully, Dreamweaver has a CSS editor that makes remembering all the different types of CSS definitions a snap. It won't allow you to create every type of valid CSS definition, but… for most

pages and sites, it's a lot quicker than hand-coding every definition from scratch. Clicking on the panel menu in Dreamweaver, you can see many of the options available with its CSS editor:

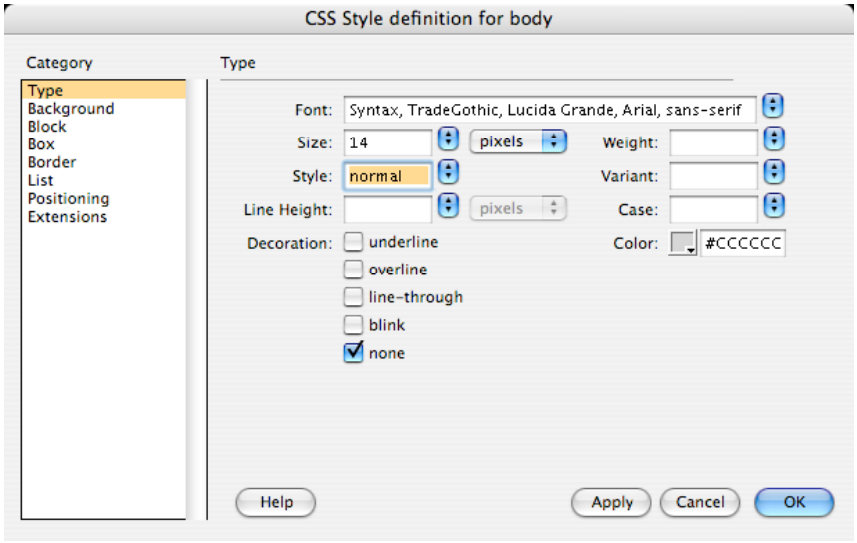


The most important part is the "Apply" and "Edit" radio buttons. Most times, I keep it in "Edit" styles. Use "Apply" when you want to apply both IDs and Classes to items in your page. It then makes it easy to apply a certain class to a number of items that would be laborious to do in code view, such as changing the "class" of "Days" in a calendar to reflect the same color. (One example, is our January lunch menu, at our web site: http://www.glnd.k12.va.us/calendars/january2004menu.html .)



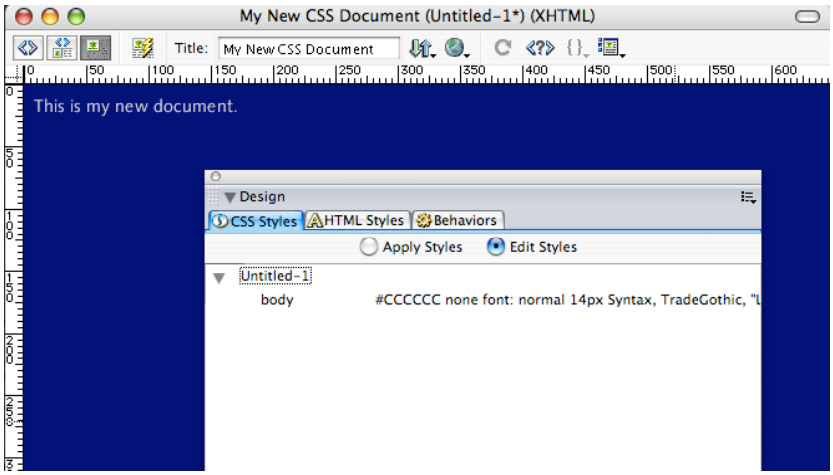
I usually start my style sheet by defining the BODY tag, so that I can choose a background, and a font size and selection that will apply to the entire page.

I then proceed and make as many definitions as I like, for the <BODY> tag for the first category, "Type":



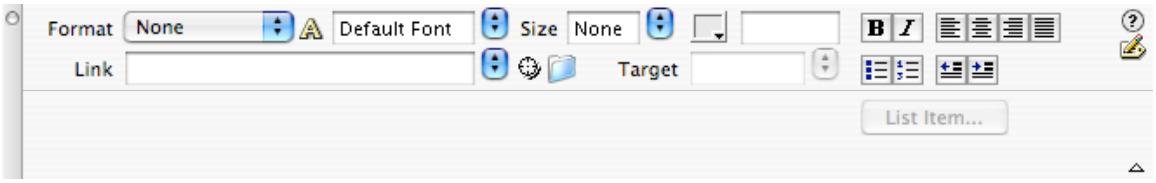
Things can get dangerous, as I navigate through the categories, if I don't know what's going on. Sadly, the further down you dig, Dreamweaver has an increasingly hard time at rendering your CSS in its Design View mode. Therefore, frequent trips to your favorite web-standards-compliant browser are commonplace. Dreamweaver MX 2003 reportedly has much better CSS support, and better rendering thanks to an Opera-based rendering engine.

Soon enough, I can begin adding content:



My CSS information is summarized in the CSS panel. By double-clicking the definition, I can make changes. Just make sure "Edit Styles" has the circle selected.

When going the web-standards route, be sure and avoid using, at all costs, the Properties Pane in Dreamweaver for manipulating fonts:



Thankfully, it does still come in handy for adding links, choosing B and I for the <STRONG> and <EM> tags, and creating lists <UL> and <OL>.

Show me the Graphics

Visual appeal is important on the web, and your school's site should have some! Photoshop, and its companion, Image Ready, are excellent tools for web design. You can begin the design process with these applications, or, simply use them to prepare graphics for your web-standards-compliant site.

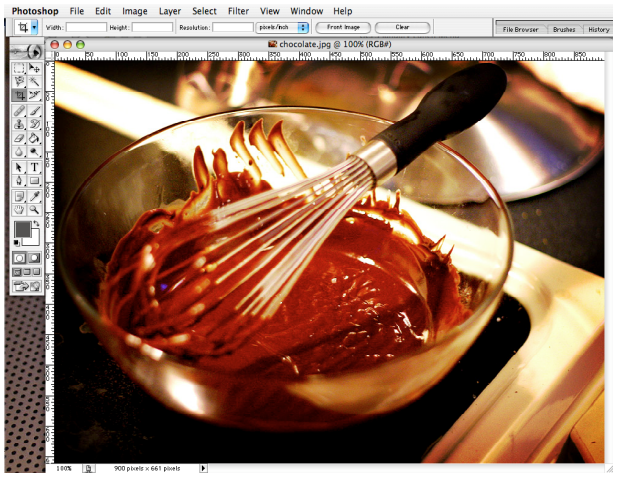
Backgrounds with Photoshop

One simple use of Photoshop is to create a background image for your website. The background can be a pattern, or a single image that stays-put in one corner of your page. Knowing the hexadecimal code for your main color is helpful when designing graphics with transparency to appear over your background—this would be your matte color.

Patterns

I won't say a lot about patterns, simply because they're typically seen as a cliché for bad designs from, say, around 1998. It was the next step, evolution-wise, from the plain-color background. Photoshop, and ImageReady, have two tools that can be used to create a custom, seamless pattern, if you just can't stay away.

Using Photoshop, we start with a picture… this is from my holiday dessert making, and brings back sweet memories.

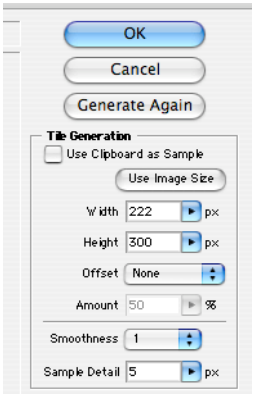


While the size of this photo (900x661) is far too large for most applications on the

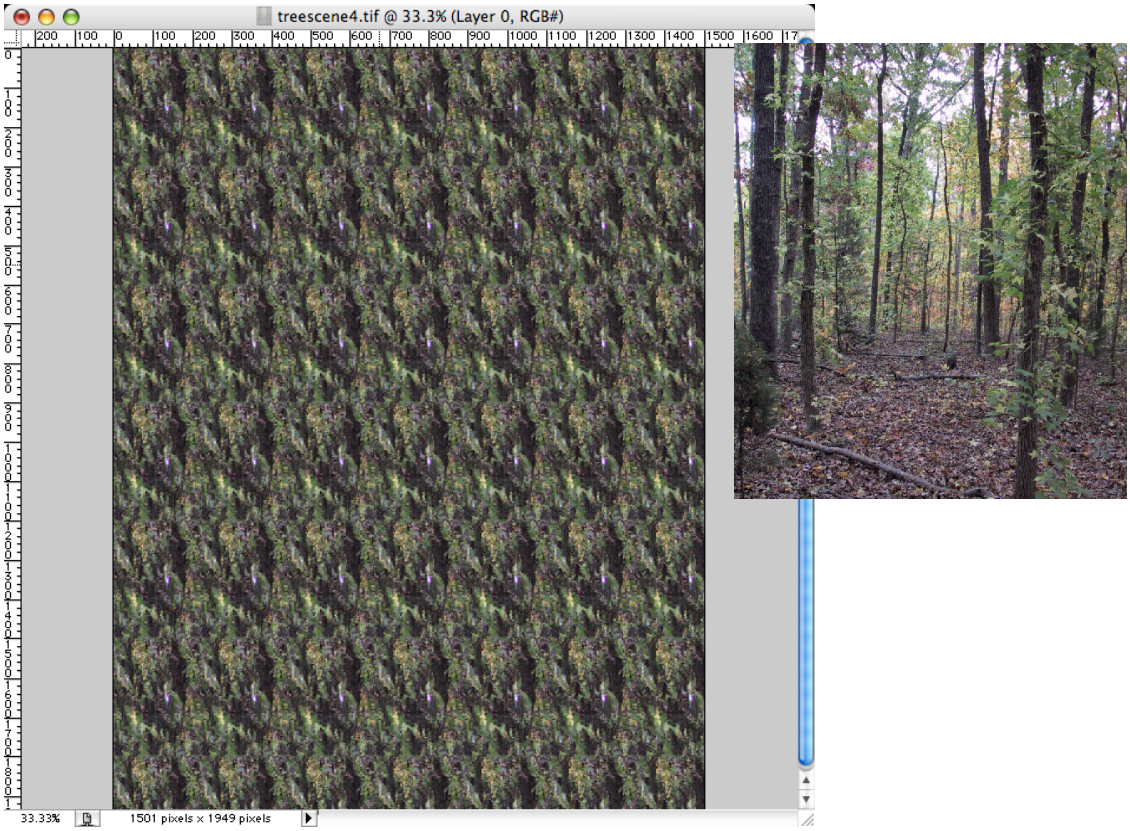
web, I can use it to develop some interesting patterns.

Under **Filter**, Choose **Pattern Maker**. Next up, is a new screen that allows you to generate, using some random variables, a pattern based on the photograph you've chosen. In my case, it's chocolate.

Choose the Generate (or Generate Again, after your first attempt) to see what Photoshop comes up with.

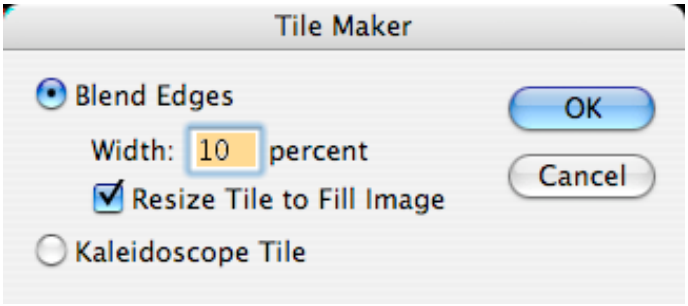


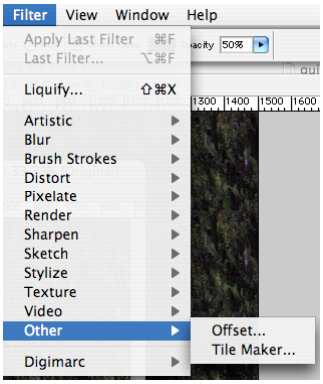
Notice you can change the width and height of your tiled piece. The picture used above didn't result in a very interesting pattern for me. This tool works best with things that are already patterns: flowers, grass, clouds, etc., by reducing the size required to fill the space. Using a scene of trees, I came up with a tile that still expressed the feeling of nature, borrowing the same sample of colors from the original:



The other Tile-maker is only found in Image Ready. To switch programs, locate the "Jump To" button at the bottom of your toolbar, and open Photoshop's companion program:

Within ImageReady, choose the TileMaker filter. I am choosing the default settings for now, and let's see what appears:





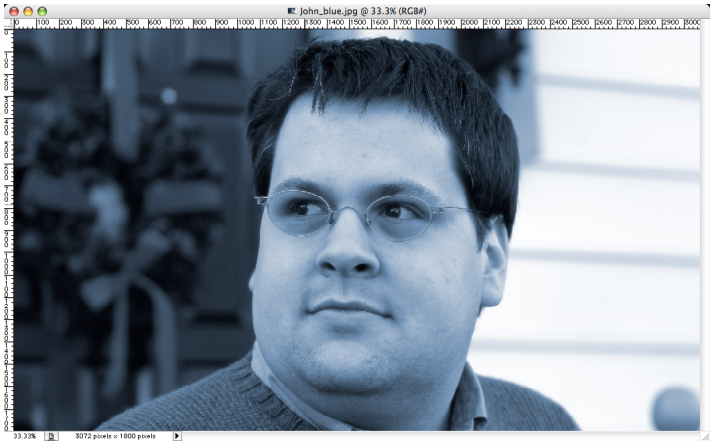
The result is a document, the size of my original, that will "blend-in" at the edges, and when repeated as a web background, will appear "seamless." In this way, you can use a small image (say 200x150 pixels) and enjoy the cost-savings for bandwidth, yet fill up an entire screen.

If my image was named tile.jpg, the CSS code would look like this:

```
body { background: #006 url(tile.jpg) repeat; }
```

Let's deconstruct! The background notation is shorthand CSS, and can be used to combine several definitions. Here, first, I define a background color of dark blue, then I also indicate I want to use a background image, which will repeat. Just leave spaces between each part. For an explanation of RGB and hexadecimal color, visit http://www.wurd.com/pwp_color.php.

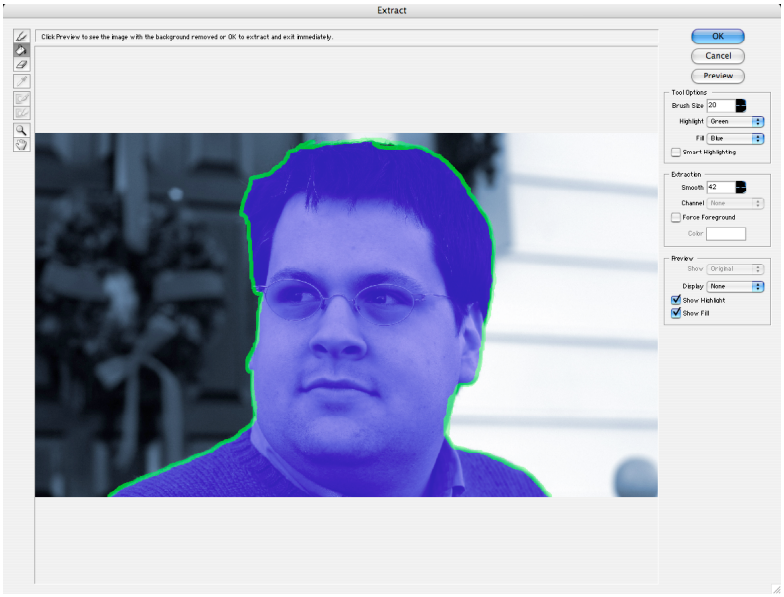
For our next background, let's try to combine several more advanced-Photoshop skills to create a background that makes use of transparency plus a static background color. The first step is to choose a photograph, and one of the author magically showed up:



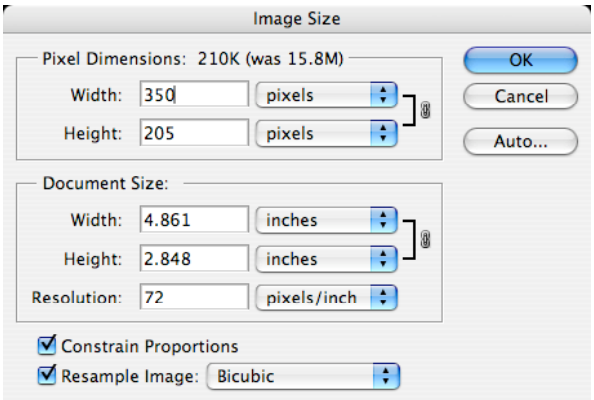
Our task is to cut out my image away from the background, and set it against a blue-ish background for the web page. First things first!

Our first step is to choose how to cut me out. Common choices include: the Freeform Magnetic Pen Tool, the QuickMask mode, or perhaps… Extract. I am going to use Extract for this demonstration. Choose **Filter > Extract**.

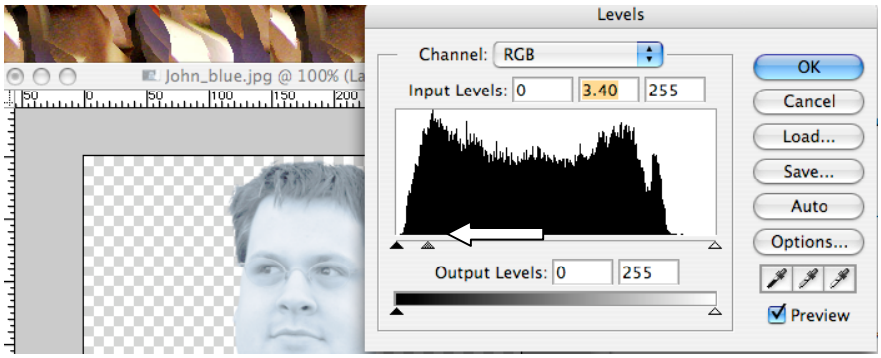
After tracing with the pen, and filling with the bucket, yours truly is ready for extraction:



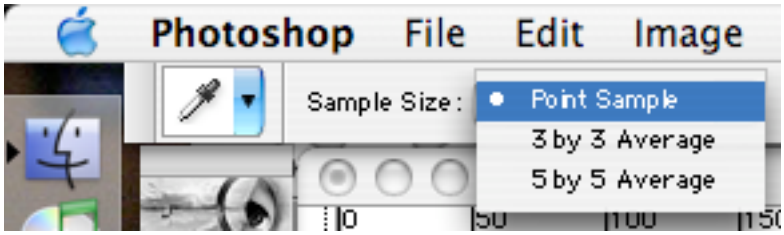
After some re-sizing, I was about ready to move into the virtual world of the Web. Make sure now, that you're working with an image at 72 PPI. This is web resolution, and every graphic you place online should be set at 72.



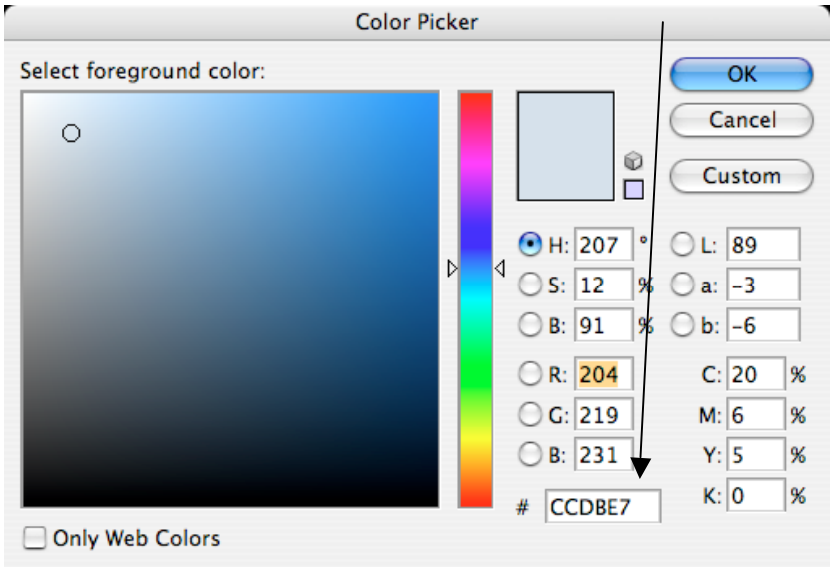
The last step is to lighten my image a bit. After all, remember this is a background, not a featured piece of the page. Switching to levels should clean things up. Try moving the midtone slider only, either to lighten, or to darken, the image. As you only move this slider, you'll reduce contrast, either way.



What's next? We need to settle on a color for our static color, and doing a sample with the eyedropper will help us out. In the options bar for the eye-dropper, I want the 5x5 sample, to get an average "blue" for my background color.

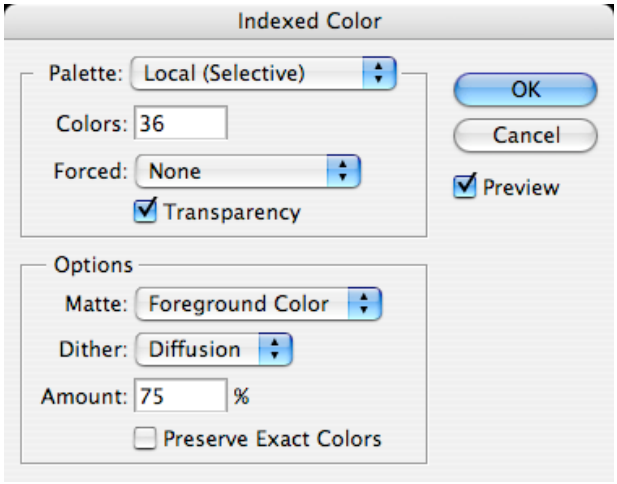


Then, using the Adobe color picker, I can see the hexadecimal value for this color:



While choosing web-safe colors is always a good idea, in this case, I didn't have a good match, and decided to live dangerously. As more of us use computers with 16-bit and 24-bit color, using web-safe colors becomes less pertinent. In fact, some have suggested using a new web-safe assortment that contains a few thousand "safe" colors. Pantone has an article of interest on the subject: (http://www.pantone.com/products/products.asp?idArticle=211&idArea=13).

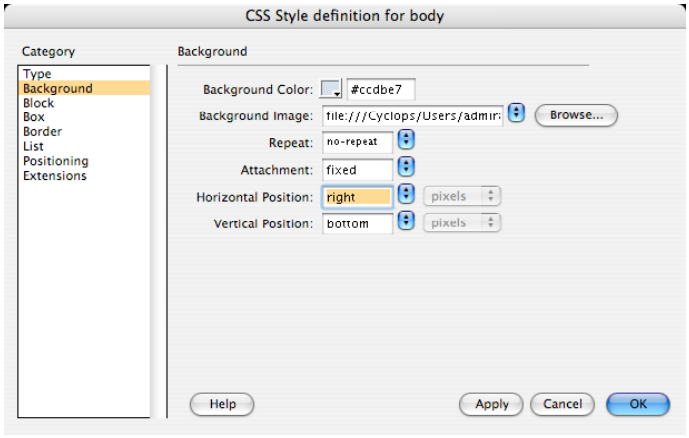
Both GIF and PNG offer us transparency, a requirement in this project. Because many copies of IE for Windows do not handle PNG properly, I am going to use GIF. PNG, however, is the superior format, and as more users upgrade their browsers, this will be great news for those of us wanting 16-bit color plus transparency. In Photoshop, choose **Image > Mode >Index**. In the dialog that appears, make your choices for colors and your matte color. If you used the eyedropper, simply select "foreground" and make sure "Transparency" is selected.



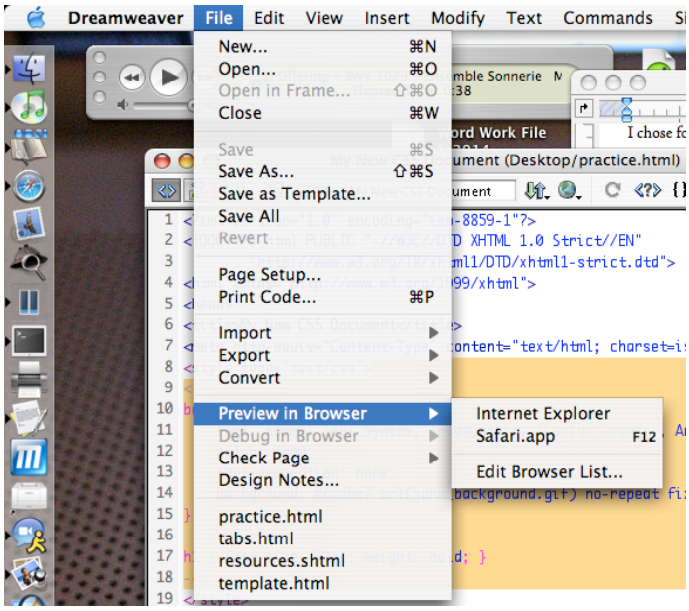
36 colors worked out great for this photograph. With Preview turned on, you can test, and as a rule, the fewer colors, the smaller your file size. Save the image as a GIF file.

Putting it Together

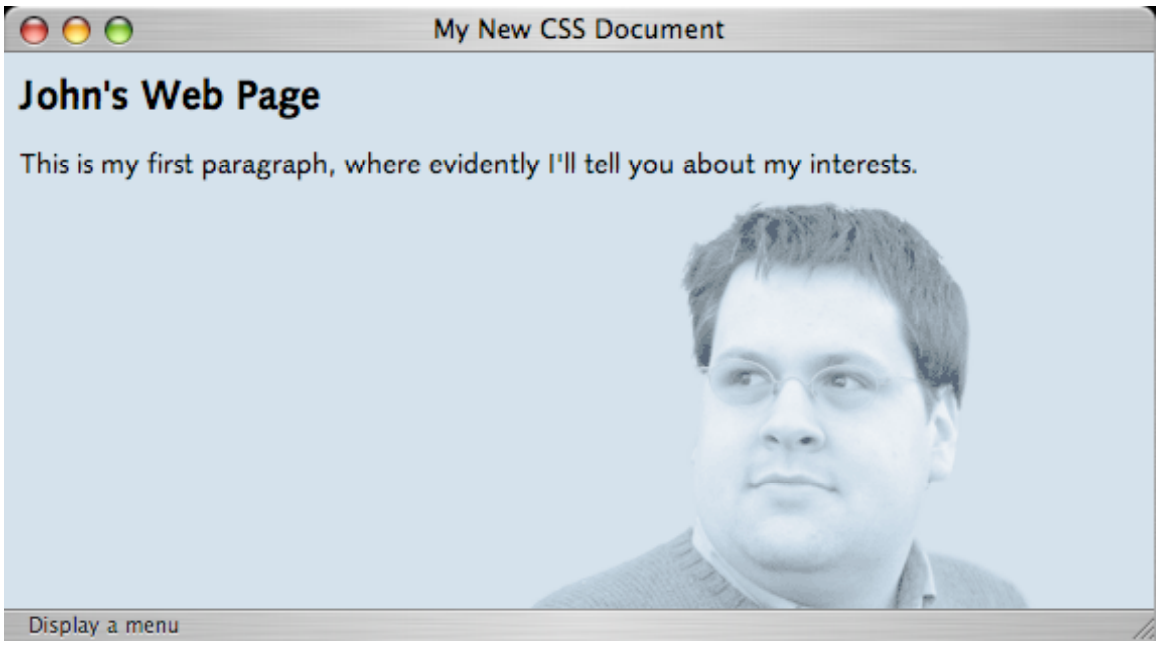
Back in Dreamweaver, I've begun a new web page, and have started to define my CSS. The settings I chose for the background reflect my design idea: I want my picture positioned in the bottom-right-hand side of the web page, and I do not want it to scroll. This is possible with CSS using the following options. Notice I used the hexadecimal color value from the Adobe color picker, in Photoshop:



Use Dreamweaver to preview your page, once you're got some CSS defined, in your favorite browser:



And the result?



The fun part about viewing this is changing the size of the window! I always obediently stay at the bottom-right of the page! Since my image is light colored, and low-contrast, even text is readable on top of my handsome photograph.

To give you a more practical example of this same technique at work, visit the Goochland County Schools homepage. In a standards-compliant browser, you should see something like the following:

Notice how the background interacts with the table beneath. The image is tacked at the top-left of the page, and blends at the right into the background color, a light gray. Because the title of the page is a GIF with transparency, it can float over the image when the page is resized.

If you really want to have fun with backgrounds, keep in mind you can also apply background images and colors to just about any element in your page with CSS, including tables, paragraphs, and <DIV>s.

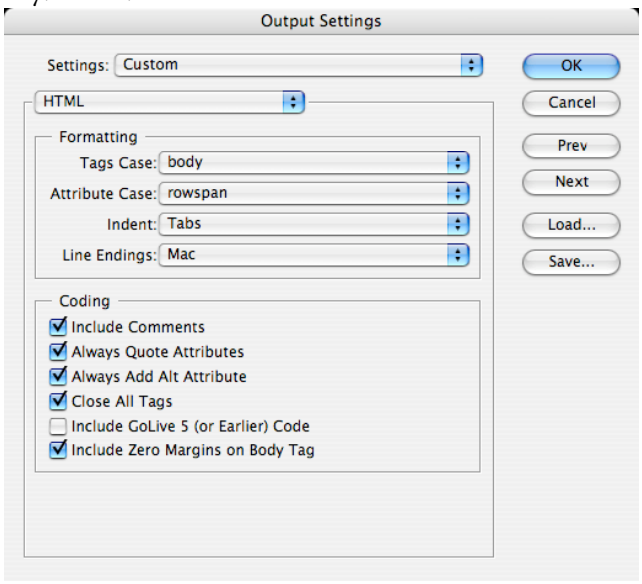
Eric Meyer (the CSS guru mentioned above), has come up with a cool way to use CSS and the way it handles background images. I have adopted his technique called Complex Spiral (http://www.meyerweb.com/eric/css/edge/complexspiral/demo.html), and used it for some pages our teachers maintain.

Using Photoshop to Design the Entire Page

While web standards evangelists have moved on to pressing XHTML and CSS to their limits with great success (Jeffrey Zeldman is one such example, and maintains a great site at http://www.zeldman.com/), some of us, used to Photoshop's power, prefer to design a page using Photoshop. This final example will layout a page which will then be sliced using Image Ready. Once sliced, you can export your creation using <TABLE>s to hold all the slices together, or, use CSS and its ability to position-by-pixel, to create a standards-compliant page. It might not have the usability and accessibility of some sites, but knowing how to massage the code and play on the strengths of your design are all part of the web design experience.

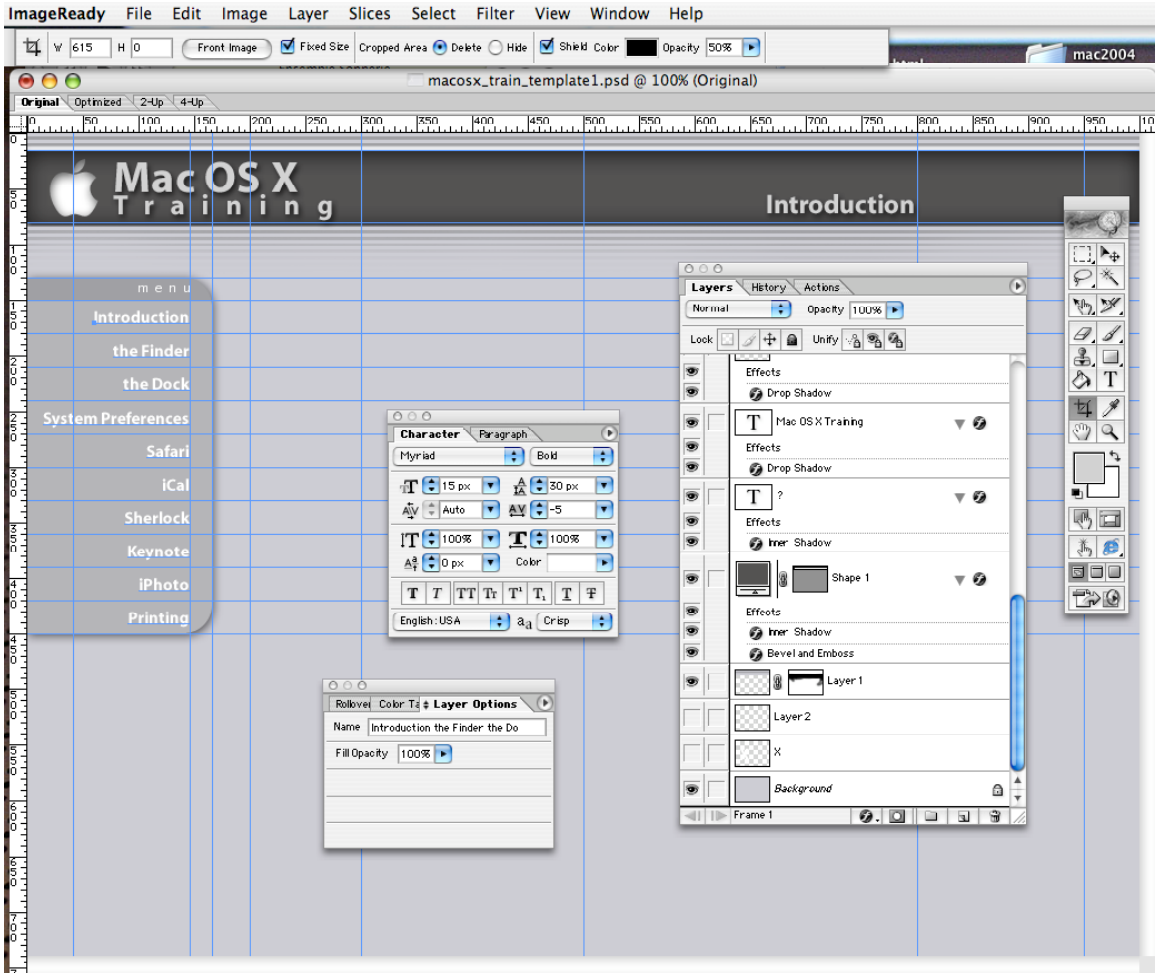
First, open a new Photoshop document. For the size, choose something larger than you'll need. I usually choose something like 1200 pixels wide by maybe 800 down with a white background. I can't tell you how to design (there are complete books and college courses for this), but I will give you a few hints:

- Make use of the guides and/or the grid in Photoshop
- Always have your rulers handy (Command/Control- R)
- Turn on the Snap-to-Grid or Guides function—it helps in positioning guides, and moving layers around
- Use the Knife/Slice tool to divide up your web page into compartments. This helps in image-loading time
- Optimize each slice for file type, quality, colors, etc.
- Turn on the options in the Output Settings that contribute to easy conversion to XHTML (in the example, opposite, I'd uncheck "Include Zero Margins…" because this should be done in CSS, not in HTML)

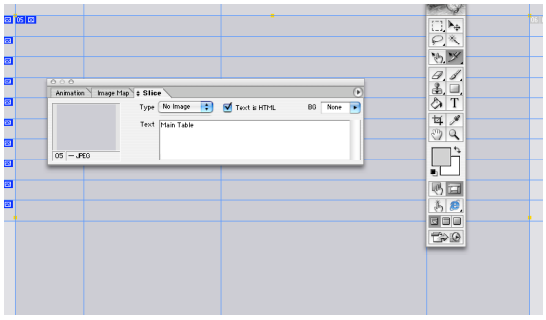


After switching to ImageReady, you can add effects such as JavaScript roll-overs to your design. Remember one basic rule, however, that simplicity should be a guiding force in your design decisions. As you add more script and effects, the file sizes of both your code and of the required graphics increases.

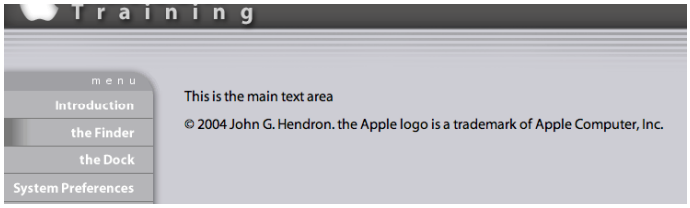
After some designing, I came up with the following design for a new layout dedicated to training.



In this close-up, notice how Slice 05 has been switched from Image to Text. This provides me with a table cell, or a CSS-defined <DIV> that I can add content to in my page, once I've switched over to Dreamweaver.



Finally, after the start of some editing, I preview the page in my browser.



Conclusion

The virtues of using Web Standards, which include, beyond CSS and XHTML, EMCA-compliant JavaScript and the document object model (DOM), are well-evangelized in the book entitled *Designing with Web Standards* by Jeffrey Zeldman (New Riders). Creating sites that adhere to standards offer, many times:

- backward compatibility with older browsers
- forward compatibility with emerging browsing technologies
- accessibility to those with disabilities and those using non-traditional browsers
- faster download times without bloated code, full of presentational markup

Lastly, the time for using web standards is "now." To quote Zeldman: "The ability to separate structure from presentation and behavior is the cornerstone of this new design approach. It is the way all sites will be designed in the future (unless they are Flash-only sites), and it is already being used on hard-core, forward-compatible sites…" (page 58—*Designing with Web Standards*, 2003 New Riders Publishing). Thankfully, two of the vendors currently trying to best-align themselves for this new approach are Adobe and Macromedia, creators of Photoshop and Dreamweaver, respectively.

As a source of parting inspiration, I might suggest perusing the CSS Zen Garden (http://www.csszengarden.com/), which highlights just what is possible using the tools we've seen here, and a new approach to designing with web standards at heart.

January 9, 2004