

Title:

Learning Cascading Style Sheets Makes For Better Web Pages.

Word Count:

994

Summary:

The other day I was asked why I like Cascading Styles Sheets (CSS) and what makes CSS so great. Here's my answer to those questions.

The Holy Grail of CSS it to completely separate web page content from the instructions that control its look and feel. If this is achieved then it's much easier for various devices to display the web page correctly. For example the same page would display correctly on a standard web browser (Internet Explorer, FireFox, Opera, Netscape, etc.),...

Keywords:

Article Body:

The other day I was asked why I like Cascading Styles Sheets (CSS) and what makes CSS so great. Here's my answer to those questions.

The Holy Grail of CSS it to completely separate web page content from the instructions that control its look and feel. If this is achieved then it's much easier for various devices to display the web page correctly. For example the same page would display correctly on a standard web browser (Internet Explorer, FireFox, Opera, Netscape, etc.), devices used by persons with a handicap, cell phones, and yet-to-be-developed interfaces. Nor would the web site designer have to make separate pages for some of these devices. Reality is quite different though and here in the real world CSS does not yet do all these things. But it does have enough positive benefits to learn how it works and to incorporate it into your web pages.

There are a different ways to control how things looks on a web page. For example, the color, size, and font used for a headline or the color, size, and font for a paragraph of text can be defined with in-line styles and tags. In-line means that these formatting commands for controlling the color, size, and font are mixed in with the content. This makes the source code for the page much harder to read and modify when you want to change it or fix a problem. In addition, because you're repeating the same commands over and over down the

page, it makes the file size of the page get larger and larger and less efficient (slower) for those browsing your site.

CSS is not repeated throughout the page. CSS can be defined in the head section of the HTML page, or put in a separate file and referenced from the HTML page, or you can even do both. CSS consists of definitions of how a page component should look on the page or device. For example, you can define that a headline should be red, 26 point, right aligned text and that a paragraph should be black, 10 point, left aligned text. Any normal HTML paragraph tags or headline tags would use these definitions when rendered.

You can define pretty much all the normal HTML objects this way; background color, background image, background image position, tables, cells, images, divs, etc. This removes the clutter from your HTML code and makes it much easier to read. But wait, there's more! If you have a web site with more than one page and you use CSS, and, you put all your CSS definitions in a separate file, you have only one place to go to change the look and feel of all the pages in your site. If you have a 50 page site and you learn that the size of your text is too small or you used the wrong color to maximize sales: instead of having to edit 50 pages and change the definition of each paragraph tag, you simply edit the CSS file and you're done!

You may be asking how to make one paragraph or a set of paragraphs look different than the default? One way is to define a class for that item. If you have a right column where you display ads, in your CSS you would make a class and give it a name such as ".rcol". You would define the necessary items for the class that you want to control (paragraph or header tags). ".rcol p" would be used to control how a paragraph tag was rendered. To associate the class to the object, simply add "class=rcol" to the paragraph tag, or the table tag if it's in a table, or div tag if it's in a div, etc. This is also where the term cascading in CSS earns its keep: the default definitions cascade down into a class as long as the class does not contain something that overrides the default. This means that in our example text rendered in a paragraph tag looks different for the rcol class. However, because that's the only thing we've defined for rcol, everything else would look the same as the rest of the page.

You can also define size and positioning for objects in CSS. This is one place where we hit the real world of CSS pretty hard. Not all browsers support the size and position commands the same way. This leads to hacks that define a position and then use a command that is known, for example, to cause Internet Explorer to bail out of the CSS, after that line you use a position command that Netscape for example understands. CSS uses the last definition of an object so this technique can be used to "trick" or "hack"

CSS into working across more browsers than it normally would. I don't recommend

doing this. One reason is that it's messy and easy to forget why you did something. The other reason is because as browsers are updated, or new devices come online, they may not follow these unwritten and unsupported hacks and your pages are apt to be all messed up. To get around this I usually use CSS as much as I possibly can and then use tables and in-line definitions to control positioning and size. Some people will go to great lengths to use CSS for everything, even replacing all tables, but here in the real world, you should get the page built, functioning, and in a form that can be used reliably on as many platforms as possible.

Not all web site software packages like Microsoft Front Page, Dreamweaver, or Adobe GoLive, etc. fully support CSS.

You'll have to do some coding manually. Don't worry, it's not that hard. I have an online course that can teach you how, just follow the link at the end of this article.

Take the time to learn and understand CSS. Implement it in your web pages. It will be time well spent.