USING THE CASCADING STYLE SHEETS FEATURE

Qiryigitov B.A.[1]

Qiryigitov Baxriddin Abdusattarovich.[1] *Teacher of department of chemistry and phyisics,*

Andijan Branch of Tashkent State Agrarian University

Andijan. Uzbekistan

Annotation: Research and study of cascading style sheets, creation of styles and work off. CSS is a language that describes the style of an HTML document. CSS describes how HTML elements should be displayed.

Key words: A personal computer, among the creatures of web applications master, the text editor MS Word.

CSS is designed primarily to enable the separation of presentation and content, including aspects such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS has a simple syntax and uses a number of English keywords to specify the names of various style properties.

A style sheet consists of a list of *rules*. Each rule or rule-set consists of one or more *selectors*, and a *declaration block*.

In CSS, *selectors* declare which part of the markup a style applies to by matching tags and attributes in the markup itself.

Selectors may apply to:

- all elements of a specific type, e.g. the second-level headers h2;

- elements specified by attribute, in particular;

- *id*: an identifier unique within the document;

- *class*: an identifier that can annotate multiple elements in a document elements depending on how they are placed relative to others in the document tree.

Classes and IDs are case-sensitive, start with letters, and can include alphanumeric characters and underscores. A class may apply to any number of instances of any elements. An ID may only be applied to a single element.

Pseudo-classes are used in CSS selectors to permit formatting based on information that is not contained in the document tree. One example of a widely used pseudo-class is :hover, which identifies content only when the user "points to" the visible element, usually by holding the mouse cursor over it. It is appended to a selector as in a:hover or #elementid:hover. A pseudo-class classifies document elements, such as :link or :visited, whereas a *pseudo-element* makes a selection that may consist of partial elements, such as ::first-line or ::first-letter.

The 3 ways to insert CSS into your web pages. The wise people who created CSS came up with 3 basic ways for you to use CSS in your web pages.

So some of you may be asking why have the 3 methods of including the CSS in a web page? The answer is: flexibility and laziness! Ok I'm kidding about the laziness, replace it with 'precision'. So what the heck does that mean?

I think the easiest way to explain to you what's going on, is by giving you real examples that demonstrate the differences. Wait a second, don't fall asleep … the examples are short and I think that once you finish, you will see how easy it really is!

Another reason that you want to continue reading this article is that you will gain a good understanding of some fundamental (and practical) CSS principles –

_____

remember that the difference between people who are really good at what they do, and those who are not so good, is in the mastery of the basics.

The heading for this part of the article hints at why you would want to go through the trouble of creating a separate CSS file instead of just typing in the CSS code in the web page itself. Can you guess? Keep trying … times up! Did you get it? I could quote you some nerd centric description that describes the advantage; the problem is that only nerds who already know would understand!

In a nutshell: by keeping the CSS code in its own file, you can link that CSS file to as many web pages as you want. This has two major advantages:

1. You will have much less code in all your HTML pages – makes the pages neater and easier to manage and makes the web pages a little faster on the download. (Although this point is really minor in most cases, and is really over blown in my opinion by some people)

2. It can potentially reduce the amount of work you have to do in a big way. Why you ask? Simple; lets say you have 50 web pages where you've set the text to be black and the headline text (text in between the <h3> tags for example) to blue. Then one day you decide you want to change the color of the text. Since the CSS that controls the text color for the 50 pages is in one CSS file, you can easily change the color of your text in all 50 pages by changing one line in the CSS file!

If on the other hand you had decided to include all your font color information in each page, you would have had to change all 50 pages. This would have been even worse if you had been using font tags.

If you are going to have more than one web page with the same stylistic properties (that look the same in some way) you should create a separate CSS file and link your web pages to it.

Method 2: Create a CSS block in the web page itself.

Use this method if you want to override the CSS you have in a linked CSS file or if you only have a one-page web site.

Now that we covered the first method of putting all your CSS code in a separate file and linking to it, the other methods are easy to describe.

_____

CSS stands for (is the acronym for): ''Cascading Style Sheets.' I think the words 'style sheets' in CSS are self-describing … we know what 'style' in style sheets mean. But what is the meaning of 'cascading' in CSS?

The cascading effect in CSS. The word 'cascading' in CSS describes a cascading mechanism; that is to say that the CSS code on the page itself will override the CSS code in a separate linked file. And subsequently, CSS declared 'inline' on the tag itself would override all other CSS.

So let's look a practical example; let's say you have a web site with 50 pages where the layout and fonts are the same on all 50 pages. Wisely you put the CSS information that sets the layout and font choices in a separate style sheet, but for a particular page you need to change the color of some of the text and add a border around a paragraph. This is a perfect example where you might want to place a little CSS in the page itself since the color and border will be unique to that page. Is this all sinking in?

Method 3: Embed the CSS right on the tags themselves (called inline CSS)
The Rule: Use this method on a unique element/tag that you want to affect with CSS.

An example can be with a special heading on the page where you want to have a little more padding than you typically do for a heading. Instead of creating a class elsewhere that will only be used on this one occasion, it makes sense to me to just include the CSS inline. I have to stress that inline CSS is something you should rarely if ever use because it can get messy quick.

Literatures

1. Stephen Schafer. HTML, XHTML and CSS. The user's Bible, 5th edition = HTML, XHTML, and CSS Bible, 5th Edition. - Moscow: Dialectics, 2011.

2. Andy Budd, Cameron Mall, Simon Collison. CSS: Professional application of Web standards = CSS Mastery: Advanced Web Standards Solutions. - Moscow: Williams, 2008.

3. Christopher Schmitt. CSS. Programming recipes = CSS. Cookbook. - St. Petersburg: BHV-Petersburg, 2007.

4. Eric A. Meyer. CSS-cascading style sheets: detailed manual = Cascading Style Sheets: The definitive Guide. - Moscow: Symbol, 2006.