ost of my friends would probably disagree with me when I say that I'm fast becoming a master of styles. Of course, they assume I'm talking about fashion and clothing. While I'm sure all of the ladies would love to see me model a Speedo™ or some form-fitting Gap™ jeans, that isn't the style I'm speaking of. Quite to the contrary, I'm talking about something that only a fellow nerd (sorry, geeks not included) could appreciate: HTML style. In particular, something called a cascading style sheet.

hat's the Hub-Bub, Bub?

So what's the big deal with these things called style sheets? Put simply, they ease the workload in maintaining the appearance of a site. Typically, you want to keep the general appearance of all the pages in a web site the same. In the past, this meant that you, the HTML programmer, were forced to modify each and every page if you ever wanted a change of appearance.

If you were working on a site with several hundred pages, this became an extremely time-consuming task. Wouldn't it be nice, then, to be able to bundle the formatting options for your entire site into one file, and then have that one file applied to each of your pages? That's the central tenet of cascading style sheets.

uilding on the Past

One of the best features of cascading style sheets is the operations that make them "cascade." As any object-oriented programmer knows, a program is made up of a hierarchy of classes. Each class inherits features from its parent or base class, and a hierarchy of inheritance is created.

Let's say that you have a file called "base.css" which describes the general stylistic features of your web pages (background image, body text, anchor formatting, and more). Perhaps for one specific page, though, you need to modify the text characteristics for body text, change the anchor formatting, and remove the background image. The changes you wish to make are simple modifications of the original style sheet, so it would make sense to have a new style sheet which inherits the properties defined in the original, but makes a few changes to the original's properties. This is where the idea of cascading comes in — a style sheet can inherit the features of another style sheet. Inheritance is set up using the following statement:

```
@import "../base.css"
```

This includes the stylistic definitions of "base.css" in the new style sheet.

eeping it Readable

Just like any kind of programming language, one needs to be able to enter comments so that down the road, the file can be modified and you can remember what each of the statements represents. Cascading style sheets (CSS) use the typical C–style comment:

```
/*   This is a comment line */
```

ollowing the Rules

The major portion of any CSS file are rule definitions. A rule boils down to a set of instructions which the web browser uses when displaying an object. Rules can be applied to links, text, tables, and nearly every other HTML object.

Each rule consists of two parts: selector and declaration. Think of a selector as the HTML keyword you wish to apply the rule to. For instance, if we want to specify a rule for the <BODY> tag, then the keyword (or selector) is BODY. Try it yourself: you want to make the background color of a paragraph blue.

he declaration specifies the parameters the browser should use when displaying the object. We'll create a simple rule (selector and definition) for the BODY of the web page:

```
BODY
     background-color:    #FFFFFF;
     font-family:         Verdana,Helvetica,sans-serif;
     color:               fuchsia;
     text-transform:       capitalize
```

The rule above sets several default properties for the <BODY> tag in an HTML document which uses this CSS. The rule states that the body of the web page will implicitly have a white background with fuchsia text. In addition, the text will be shown in all capitals, despite the capitalization in the HTML source. Finally, the font-family item tells the browser to first attempt to use the "Verdana" font to display the text. If "Verdana" is not available, then "Helvetica" should be used. If "Helvetica" is not available, use the default sans-serif font.

You may not want to use the fonts you find on your Mac, since there may not be a font of the same name on other people's computers. There are several standard font family names defined:

Family Name	Example
serif	Times
sans-serif	Helvetica
cursive	Zapf-Chancery
fantasy	Western
monospace	Courier

imple Example

Our example code this month is really simple.   First, create a text file named "test.css" and enter the BODY rule we defined above. In the same folder as your "test.css" file, create another text file called "test.html." This file should contain the following HTML code:

```
<HTML>
<HEAD>
     <LINK REL=STYLESHEET TYPE="text/css" HREF="test.css">
     <TITLE>Test of CSS</TITLE>
</HEAD>
<BODY>
     This is a test.   The text should take on the attributes
     we assigned in the "test.css" file.
```

```
</BODY>
</HTML>
```

Open "test.html" in your web browser and see how it looks! Just make sure your web browser supports style sheets...

inal Notes

If you want to get a leg-up on the whole style sheet scenario, take a look at the CSS tutorial designed by Western Civilisation:

http://www.westciv.com/style_master/academy/tutorials.html

In the past, I've always been very keen on doing all of my HTML coding by hand without any of the WYSIWYG page editors. Though I won't go into the issues which make me do my HTML that way, I will say is that as far as cascading style sheets are concerned, there's a great program called Style Master which does a fantastic job of creating and editing .css files. See Erik's review in the Nov/98 edition of Apple Wizards for more information on Style Master, or drop by a small sample site at http://warehouse.applewizards.net/ .

Jeff Frey
jeff@applewizards.net