

# From HTML to CSS

BY JAROSLAV MOHAPL

## Abstract

Part one of a cascading style sheet or CSS tutorial. CSS replace the depreciating HTML element attributes. They are used to modify display of HTML elements. This part discusses the basic syntax and style rules.

**Key Phrases** Introduction to cascading style sheet, introduction to CSS, CSS style rules, selector, property, value.

**Key Words** CSS, cascading style sheet, selector, property, value.

## Introduction

CSS is the commonly used abbreviation for Cascading Style Sheets. It can be described as a language instructing Internet browsers how to display particular HTML elements. It was developed to replace the depreciating HTML element attributes. Like attributes, CSS are used to modify display of the elements, but offer more flexibility. This part of CSS tutorials introduces the basic syntax and style rules needed for writing of a reasonably formatted web page.

## CSS Building Blocks

The main CSS building blocks are style rules. A style rule is a keyword followed by a colon and a specific value, terminated by a semicolon. Examples of a rule are: `color:white; left:20px; font-family:arial;`. The style rules can be applied at different levels and it is the level approach described next that gives CSS their generality and power.

## Element Level

The lowest level at which style rules can be applied is the element level. Let us recall that most HTML elements have form

```
<tag attribute="..."> ... </tag>,
```

where tag is a keyword, such as head, body, img, hr and so on, while attribute encapsulates properties, for example width, height, align, color etc. The appearance of elements with style rules does not differ substantially from elements with attributes, because the style rule replaces the attribute. A typical element with style rules has form

```
<tag style="property1:a; property2:b; ..."> ... </tag>.
```

The next code snippet illustrates how a paragraph attribute forcing centering of the text (first line) can be replaced by a style attribute (second line):

```
<p align="center"> ... </p>,  
<p style="text-align:center;"> ... </p>
```

The following style can be used to create justified paragraphs of indented italicized text:

```
style="font-style:italic;text-indent:20px;text-align:justify;"
```

## Class Specific Level

As the last example illustrates, a style can quickly turn into a long chunk of code. This drawback can be eliminated by moving the styles into the style element and by grouping the style rules using so-called selectors. Using the style element and a selector, the earlier paragraph example can be written as follows:

```
<html>
  <head>
    <style>
      p.indented
      {
        font-style:italic;
        text-indent:20px;
        text-align:justify;
      }
    </style>
  </head>
  <body>
    <p class="indented">
      ...
    </p>
  </body>
</html>
```

Notice that the attribute class replaces the attribute style and that is why we talk about class specific rules applied on a class specific level. In the example, the selector name indented, which is chosen arbitrarily, just to describe the newly created class of rules in the curled brackets, is preceded by

a period and the paragraph tag. The tag restricts use of the indented style rules to paragraphs only and makes them ignored, if they appear as attributes elsewhere. The same tag can be used with a number of different selectors. The next set of styles produces a red, a green and a yellow paragraph.

```
<html>
  <head>
    <style>
      p.r{color:red;}
      p.g{color:green;}
      p.b{color:blue;}
    </style>
  </head>
  <body>
    <p class="r">A red line.</p>
    <p class="g">A green line.</p>
    <p class="b">A blue line.</p>
  </body>
</html>
```

### **General Class Level**

On numerous occasions it is desirable to give the same text color not only to paragraphs but also to headlines and other kinds of text, in which case it is more advantageous to use a general class level selector. This is accomplished by simply removing the tag in front of the period preceding the selector name. The next example uses a general class level selector mblue.

```
<html>
  <head>
    <style>
      .mblue{color:midnightblue;}
    </style>
  </head>
  <body>
    <h1 class="mblue">Rhapsody in Blue</h1>
    <p class="mblue">What a night!</p>
  </body>
</html>
```

## Global Level

On the global level we eliminate the attribute class entirely by applying the style sheet rules to the tag of choice directly in the style element. Next example causes every paragraph to have italicized, indented text.

```
<html>
  <head>
    <style>
      p{
        font-style:italic;
        text-indent:20px;
        text-align:justify;
      }
    </style>
  </head>
  <body>
```

```
<p>  
    ...  
</p>  
</body>  
</html>
```

### External Rules

External rules are any of the earlier mentioned rules imported into the document from separate files using the link element. Style files have commonly the extension .css. One style file can be applied to a set of different documents, which is important for creation of larger web sites. Like the style element, the link element is also enclosed in the header. A typical example of a link element with a style file in the same folder as the HTML file looks like this:

```
<link rel="stylesheet" href="stylefile.css"/>
```

A good habit is labeling of the first and last lines in the css file using

```
/* start css.sty */  
    ...  
/* end css.sty */
```

By the way, the pair `/* */` is the CSS comment and can be used to disable one or more lines of style rules.

## References

C. D. Knukles and D. S. Yuen: Web Applications, Concepts & Real World Designs. *Wiley* 2005.

Cascading style sheets, level 1 recommendations by WWW Consortium. *World Wide Web Consortium* 1996.

Cascading style sheets, level 2 recommendations by WWW Consortium. *World Wide Web Consortium* 1998.

Syntax of CSS rules in HTML's style attribute. *World Wide Web Consortium* 2002.

W3C CSS Validation Service. *World Wide Web Consortium* 2004.