

## Experiment No. 1

**Problem: Implement html program for stylesheet consisting the use of external, internal and inline stylesheet, overriding sequences among them and use of id and class.**

Illustration:

First consider an example of HTML document which makes use of <font> tag and associated attributes to specify text color and font size –

**Note** – The *font* tag deprecated and it is supposed to be removed in a future version of HTML. So they should not be used rather, it's suggested to use CSS styles to manipulate your fonts. But still for learning purpose, this assignment will work with an example using the font tag.

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML CSS</title>
  </head>
  <body>
    <p><font color = "green" size = "5">Hello, World!</font></p>
  </body>
</html>
```

Now re-write above example with the help of Style Sheet as follows –

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML CSS</title>
  </head>
  <body>
    <p style = "color:green; font-size:24px;" >Hello, World!</p>
  </body>
</html>
```

You can use CSS in three ways in your HTML document –

- **External Style Sheet** – Define style sheet rules in a separate .css file and then include that file in your HTML document using HTML <link> tag.
- **Internal Style Sheet** – Define style sheet rules in header section of the HTML document using <style> tag.
- **Inline Style Sheet** – Define style sheet rules directly along-with the HTML elements using **style** attribute.

#### External Style Sheet

If you need to use your style sheet to various pages, then its always recommended to define a common style sheet in a separate file. A cascading style sheet file will have extension as .css and it will be included in HTML files using <link> tag.

#### Example

Consider we define a style sheet file **style.css** which has following rules –

```
red {  
    color: red;  
}  
  
.thick {  
    font-size:20px;  
}  
  
.green {  
    color:green;  
}
```

.Here we defined three CSS rules which will be applicable to three different classes defined for the HTML tags. I suggest you should not bother about how these rules are being defined because you will learn them while studying CSS. Now let's make use of the above external CSS file in our following HTML document –

```
<html>  
  <head>  
    <title>HTML External CSS</title>  
    <link rel = "stylesheet" type = "text/css" href = "/html/style.css">  
  </head>  
  
  <body>
```

```
<p class = "red">This is red</p>
<p class = "thick">This is thick</p>
<p class = "green">This is green</p>
<p class = "thick green">This is thick and green</p>
</body>
</html>
```

### **Internal Style Sheet**

If you want to apply Style Sheet rules to a single document only, then you can include those rules in header section of the HTML document using <style> tag.

Rules defined in internal style sheet overrides the rules defined in an external CSS file.

Example

Let's re-write above example once again, but here we will write style sheet rules in the same HTML document using <style> tag –

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML Internal CSS</title>

  <style type = "text/css">
    .red {
      color: red;
    }
    .thick{
      font-size:20px;
    }
    .green {
      color:green;
    }
  </style>
</head>

<body>
```

```
<p class = "red">This is red</p>
<p class = "thick">This is thick</p>
<p class = "green">This is green</p>
<p class = "thick green">This is thick and green</p>
</body>

</html>
```

### **Inline Style Sheet**

You can apply style sheet rules directly to any HTML element using **style** attribute of the relevant tag. This should be done only when you are interested to make a particular change in any HTML element only.

Rules defined inline with the element overrides the rules defined in an external CSS file as well as the rules defined in <style> element.

Example

Let's re-write above example once again, but here we will write style sheet rules along with the HTML elements using **style** attribute of those elements.

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Inline CSS</title>
  </head>
  <body>
    <p style = "color:red;">This is red</p>
    <p style = "font-size:20px;">This is thick</p>
    <p style = "color:green;">This is green</p>
    <p style = "color:green;font-size:20px;">This is thick and green</p>
  </body>
</html>
```

**Overriding styles:**

As in, there are inline styles on some markup that you absolutely can't remove, but you need to override what those styles are. This could be markup that is being inserted onto the page from foreign JavaScript or perhaps generated from the bowels of a CMS that you cannot control easily.

**HTML**

```
<div style="background: red;">
```

The inline styles for this div should make it red.

```
</div>
```

**CSS**

```
div[style] {
    background: yellow !important;
}
```

**Ex. Problem:**

**Implement html program to develop a web page using a stylesheet consisting the use of external, internal and inline stylesheet, overriding sequences among them and use of id and class.**

**Solution: Steps to create an webpage:**

1. Understand HTML, CSS and JavaScript thoroughly.
2. Pick an authoring tool. Use Dreamweaver if you can afford. Otherwise, find a free HTML text editor (such as NotePad++, Sublime). For programmers, NetBeans/Eclipse are good choice for HTML/CSS/JavaScript as they perform syntax checking and provide auto-complete.
3. Design and organize your page. Decide on the *look and feel* of your website. How many columns? What are the major sections (e.g., header, navigation menu, main content, sidebar, table of content, footer)? What is your theme (colors, fonts)? And so on.

[Note:Take a close look at your favorite websites!!! Use Firefox plugin "Firebug" or the built-in "Web Developer Tools" to inspect HTML/CSS of your favorite websites. Alternatively, you can use a CSS framework (I recommend Bootstrap) to *jump-start* your design.]

- a. Start with an initial design. Create basic layout of the HTML document

b. Partition your web page into logical section via <div> (or HTML5' <header>, <footer>, <section>, <nav>), such as header, content, footer. Assign an id to <div> that is unique (e.g., "header", "footer". Assign a common classname to sections (non-unique) that share the same style (e.g., "entry", "side-note").

c. Insert lists and hyperlinks.

d. insert table with 2 columns and 2 rows, also insert image.

e. Write the CSS *id-selectors* and *class-selectors*(e.g., #header *tag-name*,... #footer *tag-name*,... #menu *tag-name*,...) for common tags (such as h1, h2, h3, p, a:link, a:visited, a:hover, a:active), in each of the <div>'s.

```

        Apply to element with id="header" and id="footer" */
        #header, #footer {
            background-color: #eee;    /* light gray, same as #eeeeee */
        }
        /* Rule 4: Apply to element with id="footer" */
        #footer {
            text-align: right;
        }

```

```

/* Rule 5: Apply to all elements having class="new" */
.new {
    color: red;
}

```

f. Write your HTML pages. You may need to modify the CSS as you go along.

g. Repeat the previous steps until you are happy with your page's look and feel, layout, and most importantly, the contents - try not to create *yet another insignificant* website.

4. Create HTML document template.

5. Debug HTML code.