HTML – Header & CSS

We have learnt that a typical HTML document will have following structure –

```
Document declaration tag
<html>
   <head>
    Document header related tags
  </head>
  <body>
    Document body related tags
  </body>
  </html>
```

This chapter will give a little more detail about header part which is represented by HTML <head> tag. The <head> tag is a container of various important tags like <title>, <meta>, <link>, <base>, <style>, <script>, and <noscript> tags.

The HTML <title> Tag

The HTML <title> tag is used for specifying the title of the HTML document. Following is an example to give a title to an HTML document –

Live Demo

```
<!DOCTYPE html>
<html>

  <head>
    <title>HTML Title Tag Example</title>
  </head>

  <body>
    <p>Hello, World!</p>
  </body>

</html>
```

The HTML <meta> Tag

The HTML <meta> tag is used to provide metadata about the HTML document which includes information about page expiry, page author, list of keywords, page description etc.

Following are few of the important usages of <meta> tag inside an HTML document –

Live Demo

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Meta Tag Example</title>
    <!-- Provide list of keywords -->
    <meta name = "keywords" content = "C, C++, Java, PHP, Perl, Python">
    <!-- Provide description of the page -->
    <meta name = "description" content = "Simply Easy Learning by Tutorials Point">
```

```
    <!-- Author information -->
    <meta name = "author" content = "Tutorials Point">
   <!-- Page content type -->
    <meta http-equiv = "content-type" content = "text/html; charset = UTF-8">
    <!-- Page refreshing delay -->
    <meta http-equiv = "refresh" content = "30">
    <!-- Page expiry -->
    <meta http-equiv = "expires" content = "Wed, 21 June 2006 14:25:27 GMT">
    <!-- Tag to tell robots not to index the content of a page -->
    <meta name = "robots" content = "noindex, nofollow">
  </head>
  <body>
    <p>Hello, World!</p>
  </body>
</html>
```

The HTML <base> Tag

The HTML <base> tag is used for specifying the base URL for all relative URLs in a page, which means all the other URLs will be concatenated into base URL while locating for the given item.

For example, all the given pages and images will be searched after prefixing the given URLs with base URL http://www.tutorialspoint.com/ directory –

Live Demo

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Base Tag Example</title>
    <base href = "https://www.tutorialspoint.com/" />
  </head>
  <body>
    <img src = "/images/logo.png" alt = "Logo Image"/>
    <a href = "/html/index.htm" title = "HTML Tutorial"/>HTML Tutorial</a>
  </body>
</html>
```

But if you change base URL to something else, for example, if base URL is http://www.tutorialspoint.com/home then image and other given links will become like http://www.tutorialspoint.com/home/images/logo.png and http://www.tutorialspoint.com/html/index.htm

The HTML <link> Tag

The HTML <link> tag is used to specify relationships between the current document and external resource. Following is an example to link an external style sheet file available in **css** sub-directory within web root –

Live Demo

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML link Tag Example</title>
```

```
    <base href = "https://www.tutorialspoint.com/" />
    <link rel = "stylesheet" type = "text/css" href = "/css/style.css">
  </head>
  <body>
    <p>Hello, World!</p>
  </body>
</html>
```

The HTML <style> Tag

The HTML <style> tag is used to specify style sheet for the current HTML document. Following is an example to define few style sheet rules inside <style> tag –

Live Demo

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML style Tag Example</title>
    <base href = "https://www.tutorialspoint.com/" />
        <style type = "text/css">
      .myclass {
        background-color: #aaa;
        padding: 10px;
      }
    </style>
  </head>
  <body>
    <p class = "myclass">Hello, World!</p>
  </body>
</html>
```

The HTML <script> Tag

The HTML <script> tag is used to include either external script file or to define internal script for the HTML document. Following is an example where we are using JavaScript to define a simple JavaScript function –

Live Demo

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML script Tag Example</title>
    <base href = "http://www.tutorialspoint.com/" />
     <script type = "text/JavaScript">
      function Hello() {
        alert("Hello, World");
      }
    </script>
  </head>
  <body>
    <input type = "button" onclick = "Hello();" name = "ok" value = "OK"  />
  </body>
</html>
```

HTML - Style Sheet

Cascading Style Sheets (CSS) describe how documents are presented on screens, in print, or perhaps how they are pronounced. W3C has actively promoted the use of style sheets on the Web since the consortium was founded in 1994.

Cascading Style Sheets (CSS) provide easy and effective alternatives to specify various attributes for the HTML tags. Using CSS, you can specify a number of style properties for a given HTML element. Each property has a name and a value, separated by a colon (:). Each property declaration is separated by a semi-colon (;).

Example

First let's 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 chapter will work with an example using the font tag.

Live Demo

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

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

Live Demo

```
<!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 –

Live Demo

```
<!DOCTYPE html>
<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 –

Live Demo

```
<!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.

Live Demo

```
<!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>
```