

# Tutorial 01 Module

## Lecture

### **HTML: The Language of the Web**

Web pages are text files, written in a language called Hypertext Markup Language (HTML). **A markup language describes the content and format of documents.** HTML allows Web authors to create documents that can be displayed across different operating systems. HTML code is easy to understand for nonprogrammers. HTML describes the format of Web pages through the use of tags.

The Web browser controls the formatting of a webpage. Advantages include portability and speed. Web pages must be able to work with different operating systems and browsers. If you had to specify the exact appearance of a page it would dramatically increase both the size of the document file and the time required to retrieve it. HTML uses style sheets, with which a Web designer can explicitly define the fonts and formatting the Web browser applies to the document, thus providing portability and speed to the loading process.

**HTML has a set of rules, called syntax,** that specify how document code is written. The syntax **must be exact** or the code will not work. Many tools have evolved that allow you to create and manipulate HTML documents. While anyone can write an HTML document in a text editor like Notepad, newer tools include HTML converters and HTML editors.

HTML is continually evolving. Newer Web pages are compliant with XHTML standards. This tutorial teaches the latest, XHTML-compliant techniques, but also refers to rules that were true in earlier versions of HTML. When a reference window in the tutorial is labeled Deprecated, it refers to a technique that is no longer recommended. All assignments will use the latest XHTML-compliant techniques.

You will find most HTML code has opening and closing **tags**. A simple Web page will have these codes:

```
<html>
  <head>
    <!--Comments
    -->
    <title>Title of Webpage</title>
  </head>
  <body> Enter information to go on the page
</body>
</html>
```

You can see that each tag has an opening with only tag brackets <> and a closing with a forward slash</>. This is a general rule and there are exceptions, but let's take it one thing at a time. Tags are not case sensitive, but the current standard is all lower case. For this class, part of your grading criteria will include lower case text for the code.

Preplanning is very important. As you go through the tutorial, pay special attention to this. When you do your final assignment, you will be submitting your proposed webpage design.

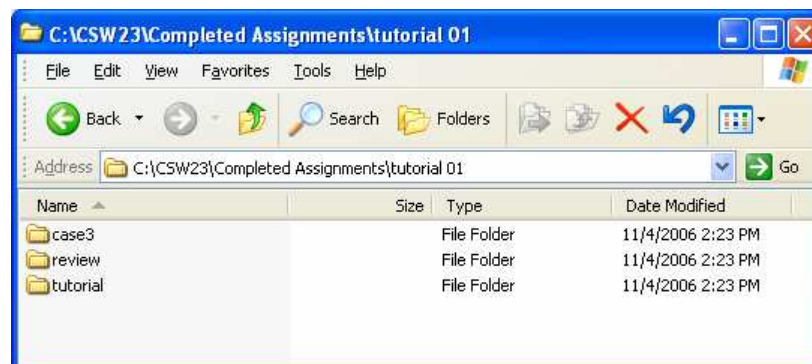
Logical tags define the nature of the tag's contents, but not necessarily how that content should appear. An example of a logical tag is the <head> tag. Physical tags define the appearance of the content. The bold tag <b> describes only the appearance of the content, but nothing about its purpose in the document.

In HTML development, the movement is away from physical tags, toward logical ones. When they have a choice, use the logical elements, and save the physical ones for situations where there is no other choice.

Go to your text book and complete the steps outlined in **Tutorial 1** through page 42. You will be asked to submit a screen shot of your folder \CSW23\Completed Assignments\Tutorial01.

You will then have 2 more assignments for this Module. The Practice/**Review** Assignment on page 44 and **Case 3** on page 50. Submit those assignments on the Assignment page of the classroom.

**NOTE:** With three assignments to do in this Module, you will want to create subfolders for each assignment inside the Tutorial01 folder. Create these three folders to complete this assignment and save all appropriate files into the folder you are working on for each assignment.



Let's get coding!

Sandi Grandberry, M.Ed.  
Wizard of Creativity  
grandberry@gmail.com