MP3 (W7,8,9): HTML Validation (Debugging) Instruction

Objectives
In this project, you will learn about:
- Explore accessibility issues and consider implications of web design on diverse audiences.
- How to validate your web pages
- How to debug

Required Readings
- Creating Accessibility Web Sites
  http://vision.unco.edu/AccessibleDesign/index.htm
- Leveling the Road Ahead: Guidelines for the Creation of WWW Pages Accessible to Blind and Visually Handicapped Users
  http://people.rit.edu/easi/itd/itdv02n4/article6.htm
- Recommended DTDs to use in your Web document
  http://www.w3.org/QA/2002/04/valid-dtd-list.html
- HTML <meta> tag
  http://www.w3schools.com/tags/tag_meta.asp

Supplemental Reading
- Fix Your Site With the Right DOCTYPE
  http://www.alistapart.com/stories/doctype/
- Web Accessibility
  http://www-03.ibm.com/able/guidelines/web/accessweb.html
- Meta Tags
  http://www.htmldog.com/guides/htmlintermediate/metatags/

Assignment
- Please complete Debugging Exercise.

I. Important Concepts

1. What is HTMLValidator?

You may not bother to make your page XHTML-compliant. Later you may find your site appearing inconsistently across the web browsers. That's why you should take the necessary steps NOW to make sure that the HTML code on your web site follows the XHTML specifications. So what is HTML validation? This is the process to analyze an HTML document in comparison to standard HTML rules, for identifying errors and non-standard codes. You can check the html validation of your web page by entering your URL at: http://validator.w3.org/

In the validator website, you can either check the html validation for your pages via their URL or you can upload your .html files to the website for checking. I encourage you to directly validate your pages via the URL so that you know the .html file on your ZipSpace is the most current.
For example, I have these codes on my page:

```html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
<title></title>
</head>
<body>
<p>This is an example of Mini Project 4 assignment. In this assignment, you need to validate:</p>
<ul>
<li>Your assignment page</li>
<li>Your Mini Project 1-4 assignment</li>
</ul>
</body>
</html>
```

Then, I validate my page on: http://validator.w3.org/, and here are the messages that I received:

This page is not Valid XHTML 1.0 Transitional. Below are the results of checking this document for XML well-formedness and validity.

1. **Line 3, Column 6: Missing xmlns attribute for element html. The value should be:**
   http://www.w3.org/1999/xhtml
   ```html
   <html>
   Many Document Types based on XML need a mandatory xmlns="" on the root element. For example, the root element for XHTML will look like:
   <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
   This error message means that I must contain specification for namespaces (xmlns) and language (xml lang="en" for English) in the <html> tag. Nowadays, this requirement also applies to the transitional DOCTYPE.
   ```

2. **Error Line 15 column 12: end tag for "ul" omitted, but OMITTAG NO was specified.**
   ```html
   </html>
   You may have neglected to close an element, or perhaps you meant to "self-close" an element, that is, ending it with "/" instead of ".
   This error message means that I have neglected a closing "ul" tag
   ```

3. **Info Line 11 column 9: start tag was here.**
   ```html
   <ul>
   This information reminds me that I have open "ul" tag that has not closed yet. Hint: All of the tags must be closed (either in pairs or self close)
   ```

4. **Error Line 15 column 12: end tag for "body" omitted, but OMITTAG NO was specified.**
   ```html
   </html>
   This error message means that I have neglected a closing "body" tag.
   ```

5. **Info Line 7 column 6: start tag was here.**
   ```html
   <body>
   This information reminds me that I have open body tag that has not closed yet.
   ```
So, here is the correction (note for the red color):

```html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title>Unit 4</title>
</head>
<body>
<p>This is an example of Mini Project 4 assignment. In this assignment, you need to validate:</p>
<ul>
<li>Your assignment page</li>
<li>Your Mini Project 1-4 assignment</li>
</ul>
</body>
</html>
```

2. What does Website Accessibility mean?
Website accessibility is becoming increasingly important since right now the website has a wider audience. Individuals with disabilities access the web in several ways. They may use customized browser settings, such as font size, and color and screen resolution. They may also rely on assistive devices such as screen readers, text readers and voice-activated devices. However, if the HTML code used to build web pages is not appropriately optimized, these assistive technologies can become ineffective, and the website can become inaccessible. An accessible web page should at least:

- provide alternative text descriptions for the images
- provide headers for columns and rows in tables that contain data
- have high contrast between background and text
- meet Section 508 Standards

II. Other Resources

The following links provide information about a variety of disability issues and gives guidelines for creating sites to accommodate all users.

**Designing Web Pages for Sight Impaired Users**
If you are involved in making Web pages accessible for users with sight impairments, check out "Disabled Accessibility: The Pragmatic Approach," by Jakob Nielsen ("The Alertbox," June 13, 1999 issue). Nielsen includes a link to the W3 Web Content Accessibility Guidelines 1.0, the W3's prioritized list of design rules, and a very useful checklist for Web page designers.

**Making Websites Accessible**
Rey Junco, counselor in the Pennsylvania State University Office for Disability Services, visited the University of North Carolina at Chapel Hill campus this month. Using examples from UNC-CH Web pages, Junco gave a presentation on creating Websites that are readable and navigable by users with sight, hearing, or motor disabilities. He also provided a list of Websites that address guidelines and the laws that cover disability issues:
II. Programming Style & Debugging

1. Programming Style
Computers and browsers don't care what your XHTML code looks like. If it is correctly structured, then everything will work fine. BUT computers are not the only things viewing your code. You must review it as you write it (looking for problems) and everyone on the Internet can view your code (viewing the source). With potentially millions of eyes viewing your code and you spending your precious time hunting for those mysterious errors, it is extremely important to write XHTML in good style. The code should be easy to read. White space and indentation are your friends. Look at the following examples.

Example 1

```html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"><html><head><title>this is a title</title></head><body><p>This is a paragraph</p><p>This is another paragraph</p></body></html>
```

Example 2

```html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html>
<head>
  <title>this is a title</title>
</head>

<body>
  <p>This is a paragraph</p>
  <p>This is another paragraph</p>
</body>
</html>
```
Both examples above are using XHTML correctly, but which one is easier to read? It is good programming practice to use example 2 to write your code. Each example will display the same way in a browser, but the first example is difficult for me to grade and difficult for you to de-bug.

When we write XHTML, we're not writing text to be viewed on a page. We're writing code that will be rendered by a browser to display stuff on a page. XHTML has structure. We do cool stuff with XHTML through the use of tags. Tags generally come in pairs: opening and closing. One of the most common mistakes of beginner, and experienced, web developers is to forget the closing tag. Get in the habit of writing your XHTML by lining up the opening and closing tags. Then all you have to do is to find the opening tag and run straight down the page to find the closing. If it's not there, then you forgot it. It's that easy!

Now you have probably noticed that some of my tags are not vertically aligned. Why? This is how I write my code. We will all write a little differently. I vertically align tags that hold other tags, but horizontally align tags that only hold text. Why? It's my style. I think it looks "pretty". :-)

The important point is to develop an indenting style and stick with it. After a while, your "style" will become such a habit that you will instinctively know when you forgot something.

2. Commenting

The comment tag (<!-- -->): When creating web pages, the page of your source code may get longer and messier when you have richer content. Using comments can serve as a reminder for yourself or other programmers about what you are doing. The comment tag (<!--This is a comment.-->) is what you need to use when adding a comments in the HTML source. The text inside the <!-- --> tag is ignored when the HTML file is parsed. Thus, the comments will not show up on web pages but only in the HTML source.

*Applying comments correctly for this assignment: In this class, we will use comments in another way. In the following projects, you will be asked to add comments for the tags you used. The purpose of adding comments in projects is to demonstrate your understanding of the tags. You will explain the functions of the tags and the reason you use them. For example, if you use a <table> tag, you will need to add comments like the following,

    <!--A table tag is used to help me control the placement of the image. The border is set to "0" so that the grid of the table will not display in the web page. -->
    <table border="0">
        <!--The <tr> tag lets me create 2 table rows. -->
        <tr>
            <!--The <td> tag is used to create columns. I have one column in the first row and 2 columns in the second row. -->
            <td>
                <!--The <img> tag is used to insert a picture of my work. -->
                <img src="3960/image.gif " alt="My hometown." width="480" height="360" />
            </td>
        </tr>
    </table>

3. The Debugging Task (Class Exercise)

Your task for this exercise is to analyze the HTMLcorrection.html file attached to the Debugging Assignment in the Assignments tool. Download the HTMLcorrection.html file and open it in your text editor such as Notepad or TextWrangler. Find the errors and fix them based on the XHTML transitional 1.0 specifications. You should not make any design decision with this file -- only fix the incorrect code. You may use the xhtml validator (http://validator.w3.org/) to help you find the bugs. Also, you can add
commenting in the source code to show which part that you have changed and the reason that you made such changes.

You’re encouraged to use the checklist of xhtml specifications provided on the page 4 of W2 Basic HTML Structure Instruction (under Content/Week 2 Folder). They should help you to interpret the error messages that you may receive from the validator.

You can use the DB: Topic 9 Weekly Discussion to ask for questions regarding the validation. When you made all necessary changes, the page rendered in a web browser should be similar to the image below.