Intro Lab

Begin with some Housekeeping

1. Create a folder on your computer to hold all the files for this class. I suggest “JavaScript Class” without the quotes. I suggest you unpack all student data files into this folder.
2. Create a subfolder for this Intro Lab. I suggest “Intro Lab” without the quotes.
3. I provided a ZIP file with your Intro Lab assignment that contains the files you will need for this lab. Extract all files to your “Intro Lab” folder.

Inspect a little JavaScript

1. Among your “Intro Lab” files, open your **MPL.htm** file in Notepad++. MAC users may use Text Wrangler or BBEdit or similar text editor.
2. Notice that this file is set up to validate to XHTML 1.0 Strict standards as are all html files in this course. We will convert ALL html files in this course to HTML5 standards. Here is what we will need to change:
   1. Notice the xml declaration in line 1.
   2. Notice the DOCTYPE declaration in lines 3 and 4.
   3. Notice the xml namespace added to the opening html tag in line 6.
3. Run the file in your favorite browser. In Notepad++, go to Run … Launch in Firefox, or your other favorite browser. Please don’t use Internet Explorer.
4. It is JavaScript that adds the numbers to the bottom of the webpage. Refresh your browser two or three times and watch the numbers change.

Validate under XHMTL 1.0 Strict

1. Go to <http://validator.w3.org>, and click the tab to “Validate by File Upload.” Browse for your MPL.htm, and click “Check.”
2. When the results display, notice the results.
   1. Note that the file was checked under XHTML 1.0 Strict. It was the DOCTYPE that told the validator that was what we wanted.
   2. Note that there is one error, and the explanation has nothing to do with the actual error.
   3. The actual error is confusion on the part of the validator. That same confusion will also exist with search engines and other html and JavaScript parsers. They are unable to tell what part of this script is JavaScript and what part of this script is HTML. This problem will be eliminated in the next step.

Convert to HTML5 Standards and revalidate

1. Let’s go back to Notepad++ (or the equivalent) and convert the file to HTML5 Standards.
   1. With HTML5, we do not need an xml declaration (currently on line #1). Remove it entirely.
   2. With HTML5, the DOCTYPE is very simple. Change the existing DOCTYPE to the following : <!DOCTYPE html>
      1. You don’t need anything else.
      2. Make sure you have the left and right angle brackets.
   3. With HTML5, we do not need an xml namespace (inside the opening html tag. Change the opening html tag to remove the xmlns and add the language declaration as follows: **<html lang=”en-US”>**
      1. Make sure you have the left and right angle brackets.
      2. We are simply declaring the language of the web page to be the U.S. version of English.
      3. By the way, if you cut and paste, the quotation marks around en-US may not work. Retype the quotation marks and you will be fine.
   4. Down in the comment section, you need to add your name and date.
      1. Where it says, “PUT YOUR NAME HERE:” do it.
      2. Where it says, “PUT THE SEMESTER & YEAR HERE:” do it.
   5. With HTML5 standards, we have to tell the browser (and the validator) that we are using “utf-8” encoding. Under the title, add the following meta tag:

**<meta charset="utf-8" />**

* + 1. Make sure you have a space before the />. That space is very important.
    2. By the way, if you cut and paste, the quotation marks around utf-8 may not work. Retype the quotation marks and you will be fine.
  1. You will need to do all of the above with every html page in the text. You may want to keep this handy.

1. Save your work. We are using ONLY the 4-letter .html extension in this class. In your previous class we used only the 3-letter extension. This process is designed to make you pay attention to extensions, an important skill in this profession. I will not accept the 3-letter .htm extension.
   1. Go to File … Save As …
   2. Verify that the “Save as type” is set to “Hyper Text Markup Language file (\*html; \*htm, …)” Always check this before renaming a file.
   3. Change the name by just adding an “l” to .htm to make it MPL.html.
2. Revalidate. You will get two errors and one warning. Most of the time, I will not care about warnings, but you must fix all errors. In this case, we will fix everything by changing all three of these issues to CSS.
   1. Below your charset meta tag you will see that there is a stylesheet link, and below that there is a set of style tags that I have provided for you. I won’t do this again.
   2. If you look at the table displayed in your browser, you will see that the border is needed round the rows only. The border being displayed is the solid border style in black. Remove the border attribute and value from the table tag. The first style we want to add is for the table rows:

**tr {border: 1px solid black}**

* 1. Save you work, check to see if the web page still displays correctly, and let’s work on cell padding. The cellpadding attribute is no longer allowed. In css, we just call that “padding.” Remove the cellpadding attribute and value from the table tag. Under the previous style, add the following style for the table data (td) cells:

**td {padding: 5px}**

* 1. Save your work, check to see if the web page still displays correctly, and let’s think about cell spacing, the space between the cells. In css, this is called border-spacing, but since we only need the borders around the rows, we don’t need this at all. Remove the cellspacing attribute and value from the table tag.
  2. Revalidate. You should get the wonderful message: “Document checking completed. No errors or warnings to show.”

1. In this assignment, you are turning in MPL.html only, BUT it MUST be zipped. Canvas does not like html code or JavaScript code. It WILL change your code if this is not done. Zip MPL.html. Make sure this is a true zip file (.zip extension). Upload your zip file to Canvas.
   1. When you need to zip files, select all the files that you need to zip together. Just hold down the CTRL key and click on each one.
   2. In Windows, right click on one of the selected files, and click on Send to … Compressed (Zipped) folder.
   3. It will name the Zip folder for one of your files. That’s not good. Change the name to something meaningful, like “IntroLab.”
   4. Upload your Zip folder to Canvas.