**Assignment 8: PHP Basics**
**INFO/CS 1300**
**Fall 2016**

---

**Overall Goals:**

In this assignment, you will learn the basic components of the PHP programming language and apply these skills to improve an existing web page. Important PHP topics covered in this assignment include functions, variables, for-loops, and using include statements.

**Tasks:**

*Web Page Subtask:*

- In Assignment 4, you designed two web pages with a fixed header. Using PHP, it is possible to consolidate repeated code using "include" statements:
    - After creating a new copy of these pages for Assignment 8, extract the HTML code for the header and save it as a separate PHP file within an "includes" subdirectory of the assignment folder.
    - Using a PHP include statement that references the header HTML file, add the fixed headers back to each of the pages.

*PHP Language Subtask:*

- Array iteration:
    - Using the starter code provided in the "q01_start.php" file, write:
        - A for loop that iterates through `$plain_array` using the variable `$i` and prints all of the array's values.
        - A for-each loop that iterates through `$plain_array` and prints all of the array's values.
        - A loop (of your choice) that iterates through `$plain_array`, tests each value to determine if it is an integer (e.g., 2) or a string (e.g., "two") and prints only the integer-valued elements of the array.
        - A loop that iterates through the associate array `$assoc_array` and prints all of its keys and values.
    - After running your code, compare its output to the output in "q1.png" and make modifications as necessary.
        - *Note:* In order to print output on new lines, you may need to end your echo statements with `<br>` tags. This is fine for the purposes of this

assignment; however, in general, `<br>` tags Are Considered Harmful and should be avoided.

- Random numbers:
    - Web development often requires consulting documentation. For this task, you must find PHP documentation online that covers the following:
        - Generating random integers
        - Appending an element to an array
    - Using the starter code provided in the "q02_start.php" file, write:
        - A function called `make_randoms` whose argument is an empty array that sets `$temp_num` to be a random number between 1 and 5 (inclusive) and appends `$temp_num` to the end of the array ten times to produce an array of ten random numbers. You may hard code this one for the 10 iterations, but this is the only exception. Then, this function should print out all of these numbers with a second loop using counter `$j`, and return the array filled with ten random numbers.
        - A function called `get_and_sort` whose argument is an empty array that calls your previous function, `make_randoms` with the empty array. Then, this function should iterate through each of the random numbers in this array using counter `$k` and printing out only the values of that array which are greater than 3. Finally, the function should return the array of ten random numbers.
        - Finally, at the end of your code file, call the function `get_and_sort`.
    - After running your code, compare its output to the output in "q2.png" and make modifications as necessary, keeping in mind that random numbers are random and may not match the values in "q2.png" exactly.
- *Note:* In both the array iteration and random numbers files, you *may not* hard-code loops for a particular array length. Even if you know the number of iterations of a loop ahead of time, you must write the code as loop.

**Deliverables:**

- New version of the site created for Assignment 4 modified according to the webpage subtask uploaded to the course server in a directory named "assignment08"
- An HTML file that codes for the fixed header within an "includes" subdirectory of the "assignment08" directory.
- Image files placed in subdirectory named "images"
- **Two** PHP files named "q01_sol.php" and "q02_sol.php" uploaded to a "php_subtask" subdirectory within the "assignment08" directory.

**Due Date:** Tuesday, November 1st at 5 p.m.

**General Assignment Policies:**

- You are free to go beyond assignment requirements, but make sure that your extra work does not interfere with those requirements. For example, adding an extra script that creates invalid HTML5 code will result in a lower grade.
- Late work will be penalized 10% per day, including holidays and weekends.
- Assume all assigned file and directory names should be used with the correct case. E.g., if the assigned file name is "index.html", "Index.html" is not acceptable.
- ***All*** media files must be credited in an HTML comment adjacent to the img or other media tag ***every*** time they are used. The credit must include a specific URL for the media file.
    - Failure to do this will result in the following penalties:
        - 25% deduction the first time.
        - 50% deduction the second time.
        - 100% deduction the third time, with possible referral for violation of Cornell's Code of Academic Integrity.
    - Note: this includes media generated by an API, e.g., an icon generated by the Facebook API.
    - Note: individual assignments may require additional credit for media files.
    - If you create a media file yourself, credit yourself by name, not by a general statement like, "I made this."
- If any of the assignment files, directory names, or variable names include one or more spaces, there will be a 5% deduction.
- The presence of unneeded files in an assignment directory that make it difficult to grade your work may result in a 5% penalty.
- You are free to find and adapt code from sources that *allow you to use the code*, but you must provide credit for the code, including a specific URL or citation for the source.
    - You may not use large blocks of unmodified code and receive full credit for the assignment. If you have any doubt about your use of borrowed code, you should consult with a TA or me.