Week 5 Concepts Assignment

*Revisions in red*

CS441

*This is exercises based on the textbook ones for Chapters 7 & 8. Note, there are additional steps or requirements that might not be in the book, or steps removed from the book, so be sure to follow this document vs directly following the textbook.*

*Note, all required student files are located in the Week 1 Module of Canvas. Look for the “Student Download files” and add to your htdocs folder. Remember that the weekly code will be copied back and forth from your local Github repository and the htdocs folder.*

*Before starting, make sure to copy the directories “book\_apps” and “ex\_starts” to your local computer in the following directory: C:\*xampp\htdocs\

**Instructions:**

# Exercise 7-1 Input and Display output

*In this exercise, you will write the PHP code that gets input from a form and displays output.*

**Open and test the application**

1. Start the Chrome browser and run the application in the ex\_starts\ch07\_ex1 directory. To do that, you can use this URL:

<http://localhost/ex_starts/ch07_ex1/>

This should display a form that has a variety of controls.

1. Enter some data, including a valid email address, and click on the Submit button. This should only display the data that you entered for the email address. *Capture a screenshot of the results:*

**Write the code that gets and displays the data entered by the user**

1. Open the index.php file for this application and review the code. Note the names that are used for the various input controls.
2. Open the display\_results.php file for this application and review the code. Note that most of the code that gets data from the controls is missing. In addition, most of the code that displays this data is missing.
3. Add the code that gets the data from the controls on the first page. Then, add the code that displays this data.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

1. For the radio buttons, display a value of “Unknown” if the user doesn’t select a radio button.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

1. For the check box, display a value of “Yes” or “No” depending on whether the user has selected the check box.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

1. For all fields that allow the user to type text into the field, make sure to convert special characters into HTML entities before displaying that data on the second page as described in figure 7-8.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

1. For the comment field, make sure to convert new line characters to <br> tags so the web page can display new line characters correctly.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

1. Test the application to make sure it works correctly. To do that, you can test text fields with special characters such as the ampersand (&), and you can press the Enter key in the comments field to enter a new line character.

***Test the solution code, does it work as expected? Any quirks? Do note, this is merely the textbook recommended solution. There are numerous ways to construct this or order the individual elements.***

***Note, Exercise 8-2 continued on the next page!***

# Exercise 8-2 Use Loops

*This exercise gives you some practice using loops.*

**Test the application**

* 1. Run the application in the ex\_starts/ch08\_ex2 directory. *(Note, ex2 not one this chapter)*
  2. Click on the Process Scores button and note that it only displays the scores and doesn’t calculate the total or the average.
  3. Click on the Process Rolls button and note that it displays the maximum and average number of rolls that it took to roll the specified number.
  4. Open the index.php and loop\_tester.php files for this application and review the code.

**Implement the score processing**

* 1. In the index.php file, add code that calculates the score total. When you’re done, clicking the Process Scores button should display the correct total and average.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

* 1. In the index.php file, use a for loop to validate the user entries instead of using multiple conditions in a hard-coded if statement.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

* 1. Test your changes to make sure they work correctly. (Capture a screenshot – **DO THIS BEFORE MOVING ONTO #8**)

**Modify the roll processing**

* 1. In the index.php file, modify the code that processes the rolls so it uses a for loop as the outer loop instead of a while loop.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

* 1. In the loop\_tester.php file, use a for loop to display the <option> tags for the drop-down list instead of hard-coding six <option> tags.

*##Document in the solution file where this code was added, (Remember to use // or /\* \*/ to comment php code), adding in the problem numbers within the citation##*

* 1. Test your changes to make sure they work correctly.

(Capture a screenshot)

***Commit this document and all edited code to your Week 5 Github repository by the deadline. Any late work must be communicated.***