CO1707 Assignment 2 Support

Introduction:

The guidance below has been prepared to support students in working towards a successful solution for Assignment 2. The worksheet breaks down Assignment 2 into three achievable 'steps', which are then associated with qualities (criteria) required for your fundamental goal ('pass' mark). For more information, please refer the assignment brief on Blackboard, including the 'marking criteria' to help in gauging progress.

Step 1: Acknowledge your goal:

The fundamental goal of Assignment 2 is to convert your HTML and JS code/ pages (client-side solution) into a server-side solution using both PHP and SQL. We are using the 'Vesta' remote resource to achieve and test this, before submitting our work to Blackboard for assessment. This essentially means that your pages must operate with a remote dataset (your database) and so must be converted to 'server-side scripting' (via PHP).

Step 2: Synthesise your objectives:

The optimum place to start/ return here is to focus specifically on the areas of the brief required for the pass mark. Specifically, any solution must demonstrate the following to be eligible for a 'high pass' (mark of 48+):

a) *All pages are defined with a .php extension,*
b) *Communicates with a MySQL database using PHP,*
c) *Presents live offers using the database,*
d) *Basic log-in functionality,*
e) *Provides a personalised greeting,*
f) *Functional user registration.*

Step 3: Apply your learning:

The following tasks (together with examples of applicable wider reading) are presented as guidance to help students achieve the above objectives, and so are considered as a minimum solution for the high-pass grade.

a) rename all .html files to .php to enable server-side operation (refer to: https://www.w3schools.com/php/php_intro.asp).

Remember: you must also update all associated file references to these pages (e.g., <a href= "index.php">. You are not limited by your Assignment 1 solution directory structure (you should add new pages, see below).

b) write (minimal) SQL to connect to a database using php 'mysqli' functionality (refer to: https://www.w3schools.com/php/php_ref_mysqli.asp).

Remember: convert your SQL result to an 'array' and check this has the correct number of records using the appropriate mysqli method… this should be one (and only one) for both validation AND verification processes!

c) write (minimal) SQL to 'read' data from the provided tables (refer to: database schema and https://www.w3schools.com/php/php_mysql_select.asp).

Remember: it is not necessary to 'convert' your JS for 'Products'/ 'Cart' in the first instance. Instead, display all records from a smaller dataset (via the homepage), maintaining / implementing 'page responsivity' here.

We have thus far discussed the tasks required for a simple 'pass' mark. That below will now focus on the more 'challenging' tasks as required, for grades beyond a pass. This requires more reading (and more work):

d) create a new page (login.php) which uses a HTML <form> to collect a set of user credentials for verification (refer to: https://www.w3schools.com/php/php_mysql_select.asp).

Remember: check your database schema carefully as part of your 'mapping' of data storage for comparison with (form) input data, as 'username' in this context may not be stored as such (consider using 'email' here).

e) extend the above to implement a 'test condition' on a defined PHP 'session' variable to acknowledge successful log-in. Hint: once verified, target 'name' field (as appropriate) from array and 'post' to index page

(refer to: https://www.w3schools.com/php/php_forms.asp and 'Week 13' lecture slides, in particular here).

Remember: store the $result of your SQL in full, including the associated (!) 'name' field of the identified record. This can be achieved using sessions (i.e., $_SESSION["name"] = (variable name goes here) ["name"]).

f) create a new page (register.php) which again uses a HTML form to collect registration details, but now combines these techniques to 'create' a record in the appropriate table (and ideally, communicates success).

Remember: there is no requirement to validate or encrypt the 'password' for the pass mark; these are 'stretch' goals which you may aim for once you have satisfied the above and have time (and energy) to push forwards.

Note: For the final task above, the following resource may provide useful reading: https://www.simplilearn.com/tutorials/php-tutorial/php-registration-form

Remember to acknowledge and reference all wider reading which has supported you, within an updated Readme file (which is also a requirement for the 'pass' criteria). This can also be captured in your comments.

It is expected that students perform (at least) one hour wider 'reading/ learning' for every hour of timetabled lab. Students should therefore 'build their work up' from the pass mark per their understanding of the topics.

Assessed (lab) demos will commence in the week following the submission deadline. Only work that has been demonstrated is eligible for grading. Students should therefore highlight these labs on their schedules.

Good luck!

End.