Assignment 3: COMP1230

Due: Friday, December 9 at midnight on GBLearn

Course Outcomes fulfilled by Assignment:

1. Define the steps to configure the environment to run the PHP application.
2. Explain the interaction between the PHP script, HTTP request, and response.
3. Use PHP control structures to solve problems.
4. Design, code, and test small PHP applications.
5. Perform various MySQL operations via PHP.
6. Develop state management for web application authentication and authorization.

Purpose

This assignment aims to convert our “Assessments” website from Assignment 2 into a fully authenticated system where *specific users* can register for an account and upload and manage their assessments. Let’s say a user, Jasmine, registers and signs into the Assessments website; she should be able to manage the assessments she previously uploaded on the uploads page, but not anyone else’s. Jasmine’s active/default Assessments records (the file she wants to use to populate the Current and Completed pages) is specific to her. These records and uploaded files are private to her account only.

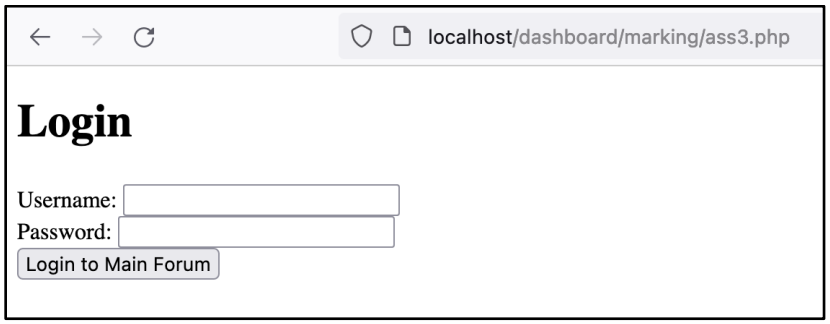
To do this, you must use an SQL database instead of the flat file structure from Assignment 2.

a. The web application should comprise of these new functionalities
   - **i. Registration**
     - To give a new user access to the Assessments system
   - **ii. Login**
     - Require login to attain authentication and authorization
   - **iii. Use of Database**
     - Use a database to manage users’ credentials and assessments records/files more efficiently
   - **iv. Upload Page Validation**
     - Ensure that data is valid first before entering new data into the system
     - For more information on the validation, check **e. Upload Page**

Functionalities

b. **Registration**
   - **i.** A new user should be able to register for the first time to your system, and their credentials should be added to a database table.
   - ii. Credentials must be **validated**
     - Username must be between 4-8 characters
     - Email must be valid (https://www.w3schools.com/php/php_form_url_email.asp)
   - **iii.** Password must be more than 8 characters

c. **Login**
   - i. You must implement **authentication** and **authorization**
   - ii. Prevents any user from going directly to a page and bypassing the login page
   - iii. Prefill the username into the login page upon revisiting the site



d. **Use of Database**
   - **i.** Use a database to store user credentials for validation (hash the password)
   - **ii.** Use a database to organize the assessments instead of using flat files

e. **Upload Page**
   - i. Perform **data validation** on an uploaded file:
     - Validate the uploaded file to ensure that it contains the correct format:
       - Values are comma separated
       - The cells contain none of the following characters: (Hint: use string functions here)
         - <
         - >
         - ?
         - *
       - There are 6 cells in each row

- The 1st cell of every row is an ID/key
- The 2nd cell of every row is 8 characters long
- The 3rd cell of every row is a string (not numeric)
- The 6th cell of every row is either “Current” or “Completed”

Technical Requirements

1. Must use a MySQL database for data storage
2. Must use browser cookies
3. Must provide adequate error messages upon failure of data validation

Best Practices

1. Make sure your code follows Best Practices. You must…
    a. Show the source code of your Assignment by using the following instructions so that you do not distort the page layout:
        i. Download: https://comp1230.gblearn.com/common/codeRetriever.zip.
        ii. Unzip the folder and drop the shs.php script in your assignment folder.
        iii. Create a link to shs.php on the footer of your main page. View Source
        iv. Open the shs.php script and update the value of the const variable "PATH_TO_ASSIGNMENT_FOLDER" to the correct path to your assignment two folder (for example, use dot for the current directory, make sure to use relative path)
        v. Example: https://comp1230.gblearn.com/common/codeRetriever/shs-link.html
    b. Each workflow in your script must have adequate comments to explain the process and logic
    c. Use proper variable naming
    d. Use indentation
    e. Use a single space before and after operators for readability.
    f. Remove unnecessary/unrelated code & comments

Submission Procedure

- **You must login to my.GBLearn.com, select comp1230 / Assignment 3 and check the plagiarism box. *No file upload is required here*. This will make an empty submission to inform me of your submission date and time and your agreement to follow academic honesty.**

- Use FileZilla to move your code to your f1231231.gblearn.com account under **public_html/assignments/assignment3**. This is the only file submission you need to do for this assignment.

- **Your submission must fully work on GBLearn otherwise, you will receive a zero.**

- Please note that plagiarism is a severe academic offence at George Brown. Students must remember that instructors can identify code that may not belong to a student, especially if it does not seem to align with their previous work. Instructors can identify their students' different styles of writing when grading academic work. Etc.). Plagiarism will always result in a grade of Zero (0) with no option to redo the assignment.

- 10% will be deducted per late day, up to a maximum of 5 late days possible.

Rubric

Question	Mark
1. Best Practices	/ 4
2. Registration a. A new user should be able to register for the first time to your system, and their credentials should be added to the registry b. Credentials validated ○ Username must be between 4-8 characters ○ Email must be valid ○ Password must be more than 8 characters ○ If a user has already been registered, don't allow the user to register for a second time. c. **Bonus:** Activate the account by sending a confirmation email to the new user's email address and providing a link to the application to update the account verification status.	/ 4 + / 4 Bonus
3. Login Module a) Look and feel (all the necessary elements are shown) b) Functionalities i) Credentials Authenticated ii) Credentials Authorized iii) Prevent any user from going directly to a page and bypassing the login page iv) Prefill the username into the login page upon revisiting the site c) **Bonus:** Lock the account for one minute after three unsuccessful attempts.	/ 4 + /2 Bonus
4. Use of Database a. Use a database to store user credentials for validation b. Use a database to organize the assessments instead of using flat files c. **Bonus:** Ability to add individual assessment modules via a form.	/ 8 + / 2 Bonus
5. Upload Page a. Perform **data validation** on an uploaded file:	/ 7

i. Validate the uploaded file to ensure that it contains the correct format: I. Values are comma separated II. The cells contain none of the following characters: (Hint: use string functions here) ● < ● > ● ? ● * III. There are 6 cells in each row IV. The 1st cell of every row is an ID/key V. The 2nd cell of every row is 8 characters long VI. The 3rd cell of every row is a string (not numeric) VII. The 6th cell of every row is either “Current” or “Completed”	
Total	/ 27 = / 15%

Note: Bonus points are added on top of your final Assignment 3 grade for a *maximum* of 100% (15/15%).