Midterm Exercise

Register and login

In this exercise, you will implement a basic registration and login system, without worrying about any styling issues. You will get practice with forms, sending requests, and working with a MySQL database.

In parts 2-3, the behavior of your page depends on whether a POST request exists. Basically, you're sending the form to the same PHP file that originally generated the form. Thus, your PHP files need to detect the existence of values in the $_POST array, and respond accordingly.

Part 1: Make the database connection

Write a PHP file that can be added to other PHP files using the include or require functions. This file should:

- Make a connection to a MySQL database, and log in with valid credentials. The connection resource should be stored in a variable with an appropriate name.
- Create a database 6470 if it does not exist.
- Select the 6470 database.
- Create a table 6470exerciseusers if it does not exist with the following fields:
  - USERNAME VARCHAR(100)
  - PASSWORD_HASH CHAR(40)
  - PHONE VARCHAR(10)
- The USERNAME field should be designated as UNIQUE.

If any of these operations cause an error, stop execution and print the error message.

Part 2: Write the registration form

Note that all of this part should be done in the same PHP file. The script should respond differently depending on the situation (whether a POST request exists, whether the username is already taken, etc.).

Write a PHP file that will output a form containing 3 fields: username, password, and phone number. These fields should be sent via POST to the same file, which should take care of inserting them into a database named 6470 and table named 6470exerciseusers (as shown above), and then confirm the registration by displaying the username and phone number back to the browser.

If the username already exists, your INSERT query should fail if you designated the USERNAME field as unique. You should query the database before attempting the insert, and if the username exists already, display an error message and a blank registration form again.

Part 3: Write the login form

Write a PHP file that will output a form containing 2 fields: username and password. Upon submission of the form, the code should check against the database to see whether the username-password pair was correct. If so, display a welcome message. If not, display the message "Invalid username or password" followed by the same login form.

Once again, there should only be one PHP file, and you should redirect to the same place after submitting. The output should be one of three options:

      1. The login form.
      2. The welcome message, if successful login.
      3. The invalid message and the login form, if failed login.

Since we haven't implemented cookies or sessions, if you successfully log in, visiting the page again (or refreshing without resending POST data) should put you back to the login form, as if you are not logged in. This is fine for now. We will fix this when we deal with session variables.