DEPARTMENT OF COMPUTER STUDIES

**(Applications Development and Emerging Technologies)**

**TECHNICAL-SUMMATIVE ASSESSMENT**

**3**

**USER INTERFACES, SESSION AND COOKIES, MYSQL DATABASE AND RESET PASSWORD FEATURE**

**Student Name / Group**

**Name: Tria, Patrick Andrew**

**Name** **Role**

**Members (if Group):**

**Section: 1A TN31**

**Professor: Mr. Abraham Magpantay**

1. **PROGRAM OUTCOME/S (PO) ADDRESSED BY THE LABORATORY EXERCISE**
   * Design, implement and evaluate computer-based systems or applications to meet desired needs and requirements.

**II. COURSE LEARNING OUTCOME/S (CLO) ADDRESSED BY THE LABORATORY EXERCISE**

* + Understand and apply best practices and standards in the development of website.

**III. INTENDED LEARNING OUTCOME/S (ILO) OF THE LABORATORY EXERCISE**

At the end of this exercise, students must be able to:

* + - To apply the available super global variables for form processing and validation.
    - To differentiate the use of $\_GET, $\_POST, and $\_REQUEST super global variable in form processing and know when to use it.
    - To differentiate the use of Session and Cookies for form security of a Web Site.
    - To know the proper syntax for validating user inputs using Regular Expression.
    - To provide a good background of Relational Database using MySQL.
    - To know the importance of Database in Web Application using MySQL.
    - To Identify the importance of Database Structure in constructing tables.
    - To be familiar with the syntax in managing users and database.
    - To define a good structure of tables in a given database for data storage.
    - To be familiar in the common syntax of creating database and tables and the correct data type to be used for each field.

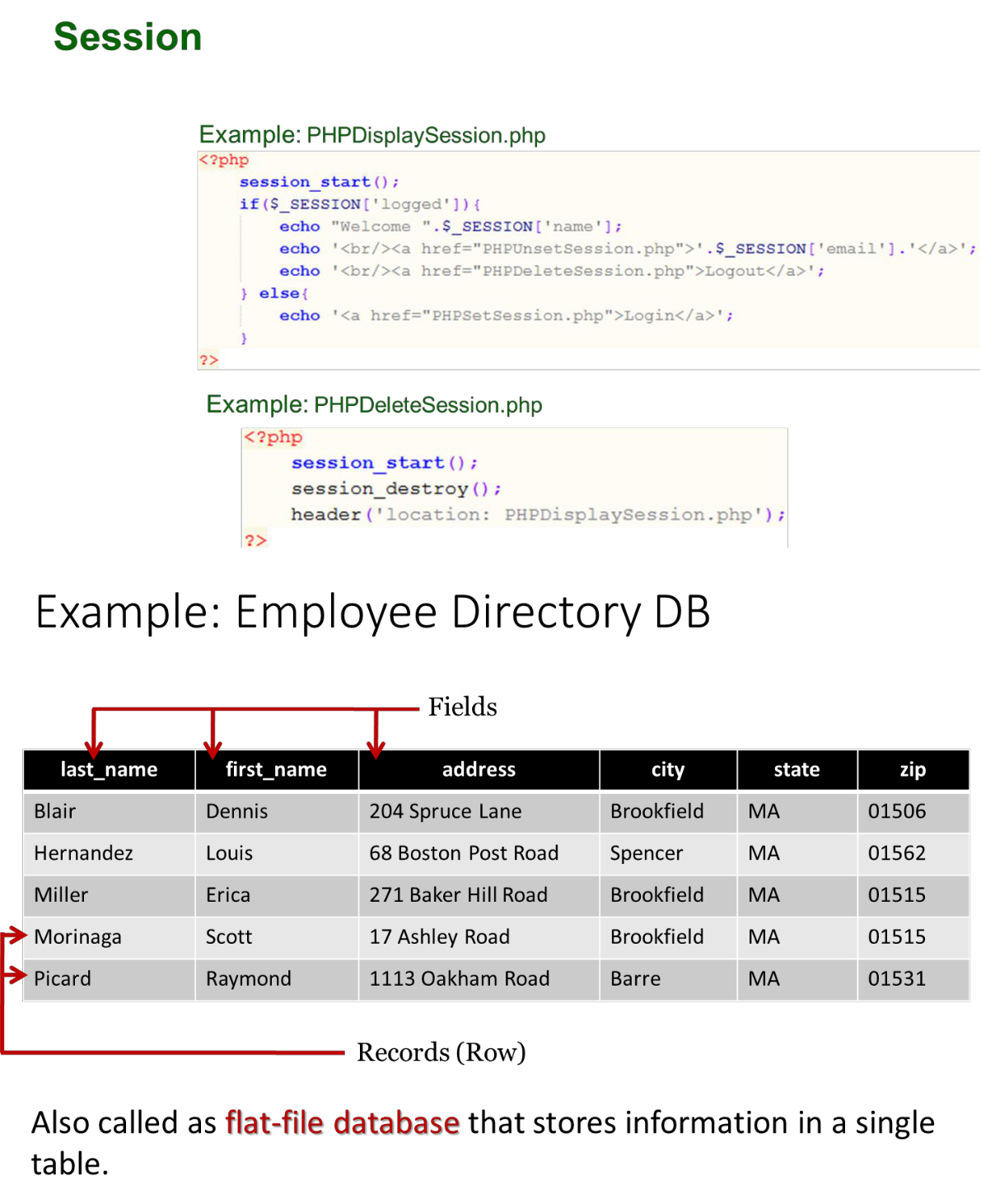
**IV. BACKGROUND INFORMATION**



**Applications Development and Emerging Technologies** **Page 2 of 11**



**Applications Development and Emerging Technologies** **Page 3 of 11**



1. **GRADING SYSTEM / RUBRIC (please see separate sheet)**

**VI. LABORATORY ACTIVITY**

**Activity A**



**Applications Development and Emerging Technologies** **Page 4 of 11**

1. Create a registration module just like what is shown below. Using $\_GET or $\_POST get the data from the forms and display below. Integrate HTML and CSS.

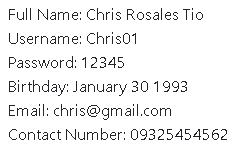


Note: the Password and Confirm Password should be the same before it show the result otherwise show “password and confirm password are not the same”.

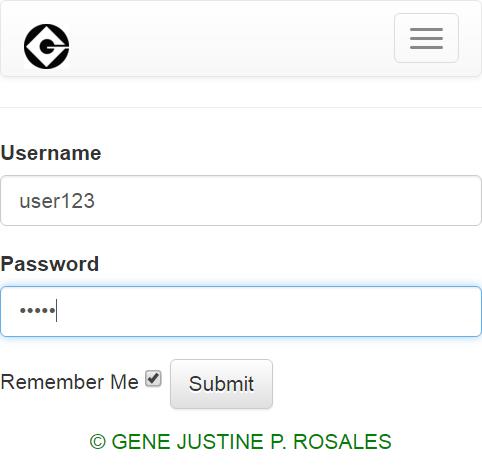
Result after click the submit button, It will show the result at the bottom of the webpage:



**Applications Development and Emerging Technologies** **Page 5 of 11**



1. Create a login module just like what is shown below. Using setcookie() function, create cookies for the username and password. It must be saved in your web form if the remember me checkbox is checked only.



1. Using the same login module, create a static variable that will hold your username and password. Get the value from the form and compare it. If the username and password matches, using $\_SESSION, redirect the user to a homepage that will display your username stored in the session and a logout link that will redirect you to a logout.php containing session\_destroy() function. If you have an active session, the user by changing the url above cannot access other pages such us the index page, login and registration pages. If there are no active sessions, users cannot open the home.php (or pages that has logout link)



**Applications Development and Emerging Technologies** **Page 6 of 11**

**Activity B**

1. Create a registration module just like what is shown below. Integrate HTML, CSS and PHP with MySQL.



Note: the Password and Confirm Password should be the same before it show the result otherwise show “password and confirm password are not the same”.

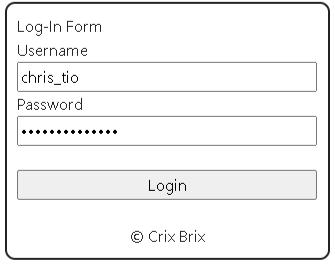
Note: You will be using this form registration to input data then save to the database.



**Applications Development and Emerging Technologies** **Page 7 of 11**

1. Using your credentials saved in your tables, it will be used to login in the system (create your own login page) include sessions in your webpages.

Example of Login page:



1. User – Side Retrieval of Record

Please see screenshot below for your guide.



**Applications Development and Emerging Technologies** **Page 8 of 11**



Notes:

* + Log-out link will destroy all sessions
  + Enter current password should be the same with the old password, otherwise show a message “Current password is not the same with the old password”
  + Enter new password and re-enter new password should be the same before it reset and save the new password, otherwise show a message “New password and Re-Enter new password should be the same.”

1. Screenshot the result of your database from the XAMPP.

***Snip and paste your source codes here. Snip it directly from the IDE so that colors of the codes are preserved for readability. Include additional pages if necessary.***



**Applications Development and Emerging Technologies** **Page 9 of 11**

**VII. QUESTION AND ANSWER**

1. **What are Super Global Variables?**

The super global variable is used to access global variable from anywhere within the PHP script. In other words, it is accessible inside the same page that defines it and as well as outside the page.

1. **Differentiate $\_GET, $\_POST, and $\_REQUEST**

The main difference between GET and POST requests is that in GET, it requests all the parameters as part of the URL wherein the user can see the parameters. While in POST, the URL is not modified at all or the parameters is hidden from the user. REQUEST is a super global variable that can catch the data which is sent using both POST and GET.

1. **What is Session?**

Session in PHP is a way to store information to be used across multiple pages. This allows you to be able to access the variables within and outside the PHP.

1. **What is a Cookie?**

Cookies in PHP are a stored on the client’s computer wherein it is kept for tracking purposes. This allows the server to be able to identify the user and fasten the process it would take to go through the different features of a website or webpage.

1. **What is the importance of validation?**

The importance of validation is so that the user won’t feel lost in your program. Proper error handling messages will make the user sure about what he/she should do. This would also ensure that the data entered is sensible and feasible, this would stop unexpected or abnormal data entering your program that might result for it to crash or receive an error.

1. **What is database?**

A database is an organized collection of data that are stored in single or multiple tables that may have relationship to each other.

1. **What is the importance of database?**

The importance of database that it stores all the important details about a certain user or information. With database, you can easily access and modify such information that is safely stored inside your computer.

1. **What is the good structure of a database?**

A good structure of a database divides your information into subject-base table to a reduced redundant data by applying the proper normalization rules. It also has the proper relationship with other tables through the use of primary and foreign key with proper data types and sizes.

**VIII. REFERENCES**

1. <https://www.w3schools.com/css/>
2. <https://www.w3schools.com/html/>
3. <https://www.w3schools.com/php/php_variables.asp>
4. <https://www.w3schools.com/php/php_superglobals_globals.asp>
5. <https://www.w3schools.com/php/php_superglobals_post.asp>
6. <https://www.w3schools.com/php/php_superglobals_get.asp>
7. <https://www.w3schools.com/php/php_superglobals_request.asp>
8. <https://www.w3schools.com/php/php_cookies.asp>
9. <https://www.w3schools.com/php/php_sessions.asp>
10. <https://www.w3schools.com/php/php_mysql_intro.asp>
11. <https://www.w3schools.com/php/php_mysql_connect.asp>
12. <https://www.w3schools.com/php/php_mysql_create.asp>
13. <https://www.w3schools.com/php/php_mysql_create_table.asp>
14. <https://www.w3schools.com/php/php_mysql_insert.asp>
15. <https://www.w3schools.com/php/php_mysql_insert_multiple.asp>
16. <https://www.w3schools.com/php/php_mysql_select.asp>
17. <https://www.w3schools.com/php/php_mysql_select_where.asp>
18. <https://www.w3schools.com/php/php_mysql_select_orderby.asp>
19. <https://www.w3schools.com/php/php_mysql_delete.asp>
20. <https://www.w3schools.com/php/php_mysql_update.asp>
21. <https://skillforge.com/how-to-create-a-database-using-phpmyadmin-xampp/>



**Applications Development and Emerging Technologies** **Page 10 of 11**

**Note: The following rubrics/metrics will be used to grade students’ output.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Program (100** | **(Excellent)** | **(Good)** | **(Fair)** | **(Poor)** |
| **pts.)** |  |  |  |  |
| **Program** | Program executes | Program executes | Program executes | Program does not |
| **execution (20pts)** | correctly with no | with less than 3 | with more than 3 | execute **(10-** |
|  | syntax or runtime | errors **(15-17pts)** | errors **(12-14pts)** | **11pts)** |
|  | errors **(18-20pts)** |  |  |  |
| **Correct output** | Program displays | Output has minor | Output has | Output is incorrect |
| **(20pts)** | correct output | errors **(15-17pts)** | multiple errors | **(10-11pts)** |
|  | with no errors |  | **(12-14pts)** |  |
|  | **(18-20pts)** |  |  |  |
| **Design of output** | Program displays | Program displays | Program does not | Output is poorly |
| **(10pts)** | more than | minimally | display the | designed **(5pts)** |
|  | expected **(10pts)** | expected output | required output |  |
|  |  | **(8-9pts)** | (**6-7pts)** |  |
| **Design of logic** | Program is | Program has | Program has | Program is |
| **(20pts)** | logically well | slight logic errors | significant logic | incorrect **(10-** |
|  | designed **(18-** | that do no | errors **(3-5pts)** | **11pts)** |
|  | **20pts)** | significantly |  |  |
|  |  | affect the results |  |  |
|  |  | **(15-17pts)** |  |  |
| **Standards** | Program code is | Few inappropriate | Several | Program is poorly |
| **(20pts)** | stylistically well | design choices | inappropriate | written **(10-11pts)** |
|  | designed **(18-** | (i.e. poor variable | design choices |  |
|  | **20pts)** | names, improper | (i.e. poor variable |  |
|  |  | indentation) **(15-** | names, improper |  |
|  |  | **17pts)** | indentation) **(12-** |  |
|  |  |  | **14pts)** |  |
| **Delivery** | The program was | The program was | The program was | The program was |
| **(10pts)** | delivered on time. | delivered a day | delivered two | delivered more |
|  | **(10pts)** | after the deadline. | days after the | than two days |
|  |  | **(8-9pts)** | deadline. **(6-7pts)** | after the deadline. |
|  |  |  |  | **(5pts)** |



**Applications Development and Emerging Technologies** **Page 11 of 11**