Text

Description automatically generated

**CENTRE FOR DIPLOMA STUDIES**

**UNIVERSITI TUN HUSSEIN ONN MALAYSIA (UTHM)**

**LAB 7**

**PHP AND MYSQL MANIPULATION WITH FUNCTION**

**COURSE CODE         DAT21303**

**COURSE NAME        WEB DEVELOPMENT**

**FACULTY                  CENTER OF DIPLOMA STUDY**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**STUDENT NAME     WAN MUHAMMAD ALIF FIRDAUS BIN WAN HANAPI**

**NUM MATRIX AA210292**

**SECTION                   6**

**LECTURE NAME    NOR NADHIA BINTI NOR AZMAN SUBRAMANIAM**

**SUBMIT DATE          8 JANUARY 2023**

**LABORATORY 7: PHP and MYSQL Manipulation with Function**

|  |  |
| --- | --- |
| **Objective** | To train the student about proper use of function manipulation in PHP. The combination of functions and MYSQL will make student become more skillful in programming skills writing |
| **Required**  **Resources** | Macromedia Dreamweaver, Sublime Text, Bracket or if you are good enough just use text editor notepad. Xampp (used to dedicate your machine as server). |
| **References** | <http://kr1.php.net/manual/en/language.functions.php>http://www.sqlcourse.com/table.html |
| **Date**  **Released** | 05/11/2017 |
| **Date**  **Submission** | 12/11/2017 |

Note: Do this Lab in a group of two people

1. **What is function?**

The real power of PHP comes from its functions; it has more than 1000 built-in functions. The function definition in programming is a named section of a program that performs a specific task. In this sense, a function is a type of procedure or routine. Some programming languages make a distinction between a function, which returns a value, and a procedure, which performs some operation but does not return a value

1. **Several type of functions:** 
   * User-defined functions
   * Function arguments
   * Returning values
   * Variable functions
   * Internal (built-in) functions
   * Anonymous functions
2. **Why use Function?**

The first reason is reusability. Once a function is defined, it can be used over and over and over again. You can invoke the same function many times in your program. Good programming writing skill where the function can be called from its name and the function can be executed from other files.

**Question 1:** By using PhpMyadmin create a new database name as **bank** and a table. The table (customers) must be created according to Table 1.0 below.

**Table 1.0: Customers**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id** | **Names** | **Password** | **Username** | **Status** |
|  |  |  |  |  |
| 1 | Gabriel | \*\*\*\*\*\* | GGabriel | Active |
| 2 | Michael | \*\*\*\*\*\* | McMichael | Active |
| 3 | Ronaldo | \*\*\*\*\*\* | RRonald | Active |
| 4 | Roberto | \*\*\*\*\*\* | RRobert | Active |

**Connection.php**

<?php

$servername = "localhost";

$username = "root";

$password = "";

$db = "majuBank1";

// Create connection

$conn = new mysqli($servername, $username, $password, $db);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

/\*echo "/\*Connected successfully <br>";\*/

?>

**Lab tujuh question 1**

<!doctype html>

<html>

<head>

<meta charset="utf-8">

<title> Customer Info </title>

</head>

<body>

<!--FORM PERSONAL INFO-->

<br>

<table border="3px" bordercolor = "black">

<tr>

<td >

<table>

<form action='labtujuhques1.php' method="POST">

<tr> <th colspan="2"> Personal Info<br><br> </th> </tr>

<tr>

<td>Names:</td>

</tr>

<tr>

<td><input type = "text" name = "names" size="42"></td>

</tr>

<tr>

<td>Password : </td>

</tr>

<tr>

<td><input type = "password" name = "password" size="42"></td>

</tr>

<tr>

<td>Username:</td>

</tr>

<tr>

<td><input name = "username" size="42"></td>

</tr>

<tr>

<td>Status:</td>

</tr>

<tr>

<td><input name = "status" size="42"></td>

</tr>

<tr>

<tr>

<td><br>

<input type = "submit" name = "save" value = "Save">

</td>

</tr>

</form>

</table>

</td>

</table>

<!--INSERT -->

<?php

include "labtujuhconnection.php"; // Using database connection file here

/\*<!--SAVE DATA--> \*/

if(isset($\_POST['save']))

{

$names = $\_POST['names'];

$password = $\_POST['password'];

$username = $\_POST['username'];

$status = $\_POST['status'];

$insert = "INSERT INTO customers (`names`, `password`,`username`,`status`) VALUES ('$names','$password','$username','$status')";

if (mysqli\_multi\_query($conn, $insert)) {

echo "New records created successfully";

} else {

echo "Error: " . $insert . "<br>" . mysqli\_error($conn);

}

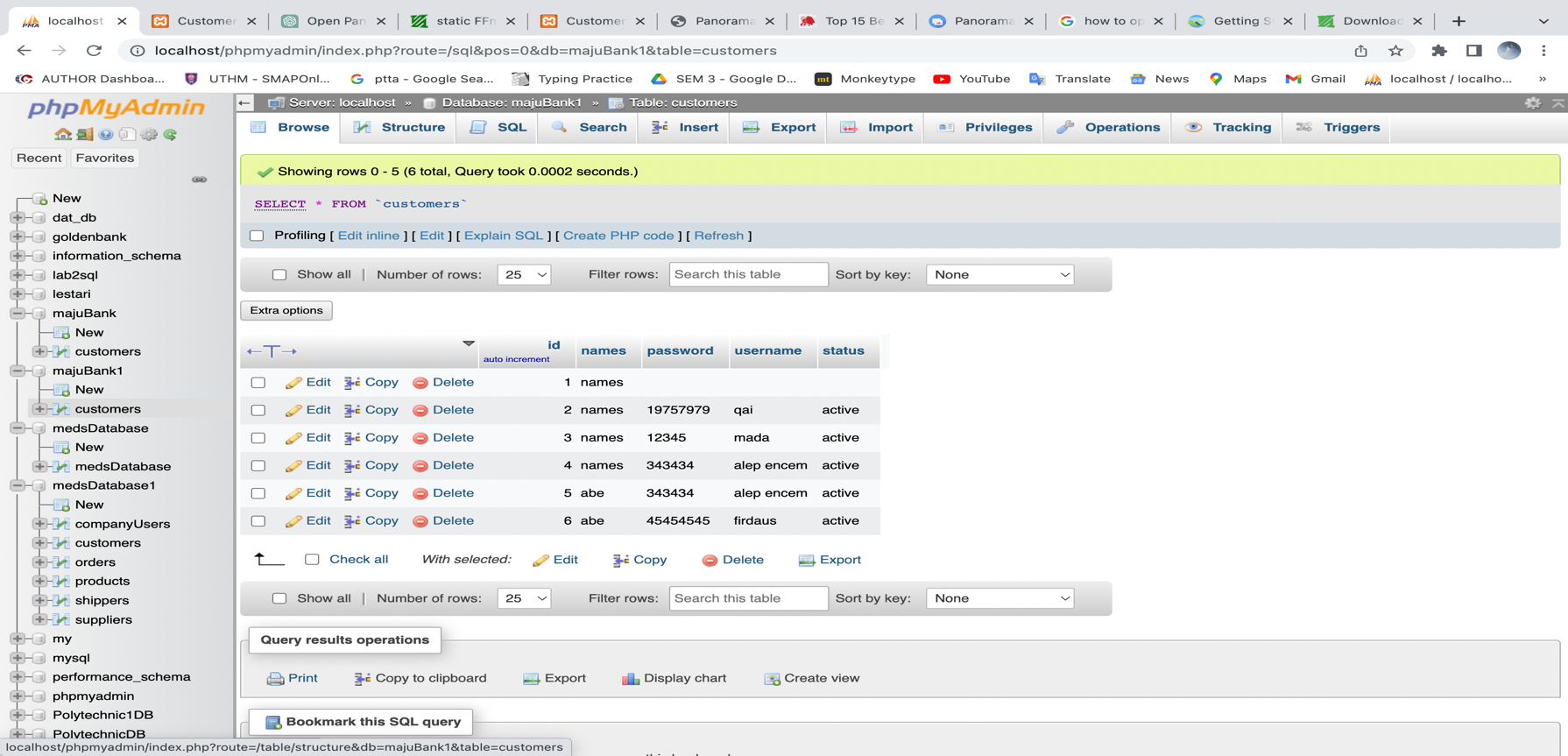
$conn->close(); // Close connection

}

?>

</body>

</html>



Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application

Description automatically generated

**Question 2:** Create a file **login.php** such in Figure 1.0 below and attach a **function** to verify the authorized user from table **customers**. Once the login successfully granted. You are required to display user name and the status such as Figure 1.1 and after **sign-out** link being clicked the session will automatically cleared. List of files must be created:

* Connection file name **config.php (for database connection)**
* Login page **login.php (for user login and session started)**
* Sign out page **logout.php (to clear the session and logout)**
* Session page **session.php (setup the session)**
* Welcome page **welcome.php (pass the value of session and display user name and status)**

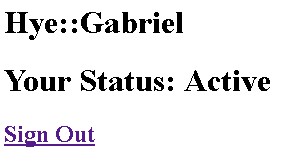
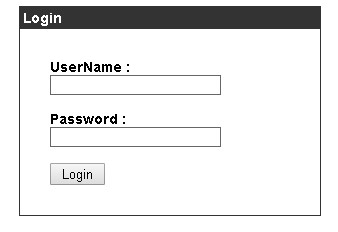
**Figure 1.**

**0**

**:**

**login**

**.php**



**Figure 1.1: Welcome.php**

**Question 3:** Explain the use of **mysqli\_fetch\_array(), mysqli\_real\_escape\_string (), isset(), session\_start(), ob\_start(), session\_destroy()** and **header().**

A result row from a result set can be retrieved using the mysqli fetch array() method as an associative array, a numeric array, or both. When given a result set, it returns a row of data in the form of an array, with the keys being the column names from the result set and the values being the relevant data.

The function mysqli real escape string() is used to generate a valid SQL string for use in a SQL statement. This is accomplished by escaping the string's special characters, like quotes, which can be exploited to insert malicious code into a SQL statement.

The function isset() is used to check whether a variable is set and is not NULL. If the variable is set, it returns TRUE; otherwise, it returns FALSE.

A new session can be started or an existing one can be continued using the session start() function. A cookie is placed on the user's computer to store the session ID as soon as a session is initiated. Data can be saved in this way and accessed by many page requests.

The function ob start() is used to enable output buffering. The term "output buffering" refers to the practise of storing output in a buffer before transmitting it all at once to the browser when the script has finished running. This can be advantageous for a variety of purposes, including modifying the output prior to sending it to the browser or delivering HTTP headers after the output has already been delivered.

A session can be terminated using the session destroy() function. The session cookie is deleted from the user's computer together with any data related to the current session.

The HTTP headers are sent to the browser using the header() method. It can be used to change the HTTP status code, reroute the user to another URL, specify the response's content type, and more.

**Psychomotor Rubric for Laboratory 7**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Criteria | Beginner (1) | Moderate (2) | Good (3) | Excellent  (4) | Scores |
| Recognize and able to demonstrate function and html object (variables) manipulation | Ambiguously presented but  available | Able to  accomplish correctly but not really clear | Presented in correctly and clear | Perfectly  clear and proper |  |
| Able to differentiate between function in basic security approach (session start, header and clear the session) while dealing with dynamic form | Ambiguously presented but  available | Able to  accomplish correctly but not really clear | Presented in correctly and clear | Perfectly  clear and proper |  |
| Able to manipulate and control variables from interface input and retrieve data from database to make a comparison | Ambiguously presented but  available | Able to  accomplish correctly but not really clear | Presented in correctly and clear | Perfectly  clear and proper |  |
| Combination of PHP function and  SQL, HTML form in proper way | Ambiguously presented but  available | Able to  accomplish correctly but not really clear | Presented in correctly and clear | Perfectly  clear and proper |  |
| Theory and Practical understanding and proper coding structure | Ambiguously presented but  available | Able to  accomplish correctly but not really clear | Presented in correctly and clear | Perfectly  clear and proper |  |
|  | |  | Total Scores (20 Marks) | |  |