**Lab 04 Server-side Programming using PHP (3)**

**Module Intended Learning Outcome :**

**On completion of the module, students are expected to be able to:**

**• use appropriate development tools to examine and develop dynamic web applications with international standards.**

**Lesson Intended Learning Outcome:**

**On completion of this lab, students are expected to be able to:**

**• Create PHP programs to collect and validate user’s input from HTML form**

**Lab4a.php:**

**1. Create lab4a.php with the following PHP code:**

<?php

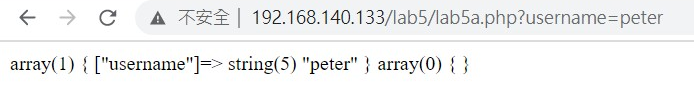
var\_dump($\_GET);

var\_dump($\_POST);

Save the file into the directory /var/www/lab4

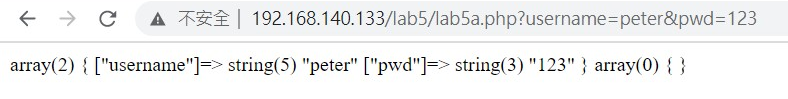
**2. GET request by URL**

Test the PHP file in the browser by <http://localhost/lab4/lab4a.php?username=peter>



Test the PHP file in the browser by

<http://localhost/lab4/lab4a.php?username=peter&pwd=123>



**Lab4b.php:**

1. Create lab4b.php in /var/www/lab4. Test the PHP file by <http://localhost/lab4/lab4b.php>
2. Create a HTML form with a textbox and a **submit** button.

<html>

<head></head>

<body>

<form method="post" action="">

<p>Username: <input type="text" />

<input type="\_\_??\_\_" name="submit" value="submit"/></p>

</form>

</body>

</html>

1. Try to click the submit button. Since the form has no ‘action’ attribute is added, form data are submitted to lab4b.php again.
2. Add the following PHP code before the <html> tag, then go back to the lab4b.php, enter username and click the submit button to submit the data:

<?php

var\_dump($\_GET);

?>

<html>

1. Can you get the data input? If you can only see empty message are displayed from var\_dump function, which mean $\_GET is not correct.According to the HTML form source code, the form method is post, it is required to use $\_POST to get the submitted data from the form, modify the request array to $\_POST and try again.

<?php

var\_dump(**$\_POST**);

?>

1. Can you get the username data inputed in the textbox? Add the ‘**name’** attribute to the textbox.

<p>Username: <input type="text" **\_\_??\_\_="username"** />

1. The result should be shown only if form data is submitted. Add the following source code to use the submit button is to detect whether it is a submission:

<?php

if( array\_key\_exists('submit', $\_POST) ) {

var\_dump($\_POST);

}

?>

1. Leave the website and visit lab4.php again. The program will not display the array data initially.
2. Modify the program to print the username and the corresponding message:

<?php

if( array\_key\_exists('submit', $\_POST) ) {

**//**var\_dump($\_POST);

echo 'Hello, ',$\_POST['username'];

}

?>

1. Enter several spaces in the textbox and click ‘submit’. Can you still see the above statement?
2. Check if the array item exists and use trim method to prevent empty string input.

<?php

if(array\_key\_exists('username', $\_POST) &&

trim($\_POST['username']) !== "" ) {

**//**var\_dump($\_POST);

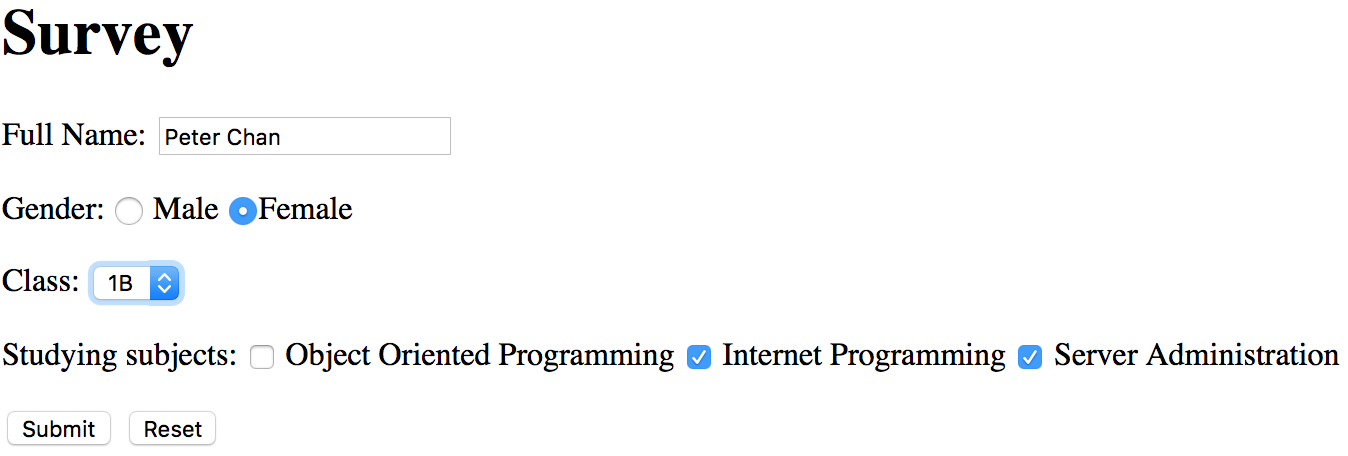
Screen capture your Lab4b output and source code and paste it on the following answer area provided:

|  |
| --- |
|  |

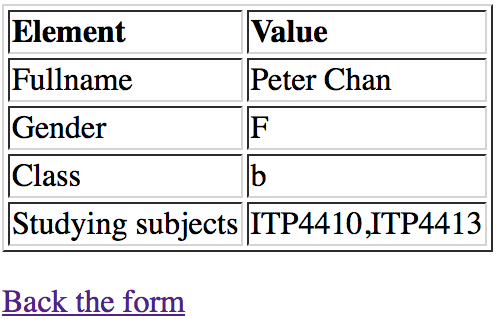
**Lab4c**

Copy file **lab4c\_form.php** and **lab4c\_submit.php** from datafile to /var/www/lab4 . Test the PHP file by http://localhost/lab4/lab4c\_form.php

Modify the lab4c\_form.php to make the program printing data input after submission as follows:



↓ ↓ ↓



A screen shot of a computer

Description automatically generated

**Lab4d**

Given the following source code of lab4d\_form.php

<html>

<head></head>

<body>

Member Registration Form

<form method="post" action="lab4d\_submit.php">

<p>Email: <input type="email" name="email"/><br>

password: <input type="password" name="password"/><br>

Date of Birth<input type="date" name="dob" /><br>

Gender:<input type="radio" name="gender" value="M"/>Male

<input type="radio" name="gender" value="F"/>Female

<br>

interest: <input type="checkbox" name='interest[]' value="Sport"/>Sport

<input type="checkbox" name='interest[]' value="Food"/>Food

<input type="checkbox" name='interest[]' value="sport"/> Video Game

<br>

address line1: <input type="text" name="address1" /><br>

address line2: <input type="text" name="address2" /><br>

<br>

<input type="submit" name="submit"/>

</p>

</form>

</body>

Create the Lab4d\_submit.php file with the following source code, complete the missing part

<?php

if( \_\_??\_\_ ) {

echo "Please double confirm your input </br>";

echo 'Email :'. \_\_??\_\_ .'</br>';

echo 'Date of birth :'. \_\_??\_\_.'</br>';

echo 'Gender :'. \_\_??\_\_.'</br>';

echo 'Address1 :'. \_\_??\_\_.'</br>';

echo 'Address2 :'. \_\_??\_\_.'</br>';

foreach( \_\_??\_\_ as $key => $item){

echo 'Your interest'.( \_\_??\_\_+1). ' is '.\_\_??\_\_."<br>";

}

}

Please enter your email address on the email input field.

Graphical user interface, text, application

Description automatically generated

Excepted output:

Text

Description automatically generated

A screenshot of a computer program

Description automatically generated

**Lab4e**

Given the source code of lab4e\_form.php and incomplete source code of lab4e\_submit.php. on data file. Please complete the following task and test case.

Task 1 Password validation

You can refer the completed source code on username validation to complete the following step:

1. Check if the password is set in POST data.
2. Trim whitespace from the password and assign the corresponding data into the variable $password.
3. Check if password is empty after trim.
4. Validate the password against the regular expression pattern.
5. Add an error message if the password field is not set at all.

Task 2 Phone validation

You can refer the completed source code on username validation to complete the following step:

1. Check if the phone is set in POST data.
2. Trim whitespace from the phone and assign the corresponding data into the variable $phone.
3. Check if phone is empty after trim.
4. Validate the phone against the regular expression pattern.
5. Add an error message if the phone field is not set at all.

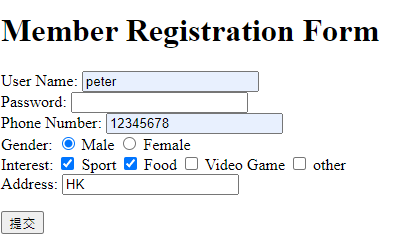
Task 3 Interest validation

You can refer the completed source code on Gender validation to complete the following step:

1. Check if the interest is set in POST data.
2. Assign the corresponding data into the variable $interest.

The following test case for you to test your program, screen capture your output and paste it on the answer area provided:

**Test case 1:**



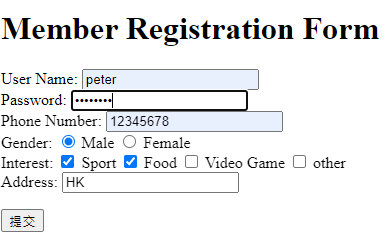
Expected output:



Test case 1 answer area:

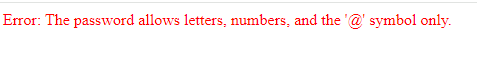
|  |
| --- |
|  |

**Test case 2:**



Enter zxcvbnm()

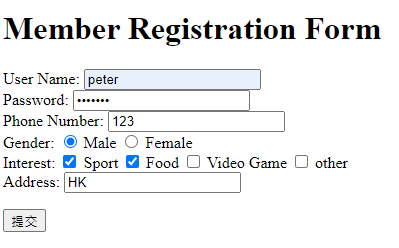
Expected output:



Test case 2 answer area:

|  |
| --- |
|  |

**Test case 3:**



Enter zxcvbnm

Expected output:



Test case 3 answer area:

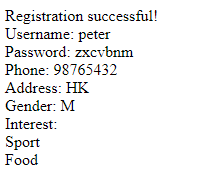
|  |
| --- |
|  |

**Test case 4:**



Enter zxcvbnm

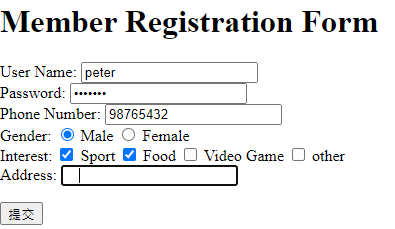
Expected output:



Test case 4 answer area:

|  |
| --- |
|  |

**Test case 5:**



Enter zxcvbnm

Enter three white spaces.

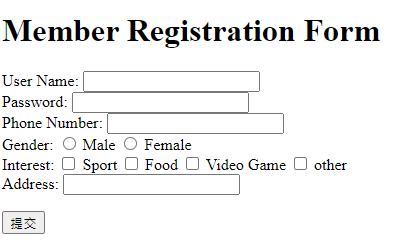
Expected output:



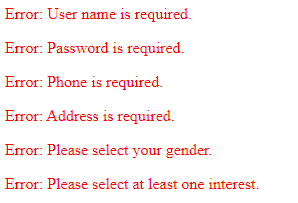
Test case 5 answer area:

|  |
| --- |
|  |

Test case 6:



Expected output:



Test case 6 answer area:

|  |
| --- |
|  |