COWB40385 Introduction to Web Programming

# Introduction to Web Programming Assessment Part 1

## Preliminary

Each exercise below should be completed as a separate web page. Once you have completed Exercise One you should make a copy of the web page you have produced and use that copy as the basis for Exercise Two. This process should be repeated until completion of the final exercise.

All web pages should use, and be compliant with, the following doctype:

<!DOCTYPE html>

All functionality should take place on the server and be written in PHP.

## Exercise One – Basic Output

Create a new PHP file (using the .php extension) and type the following code block at the beginning:

<?php

$title = "Mr";

$name = "Philip Windridge";

$email = "p.c.windridge@staffs.ac.uk";

$address[0] = "The Ocatagon";

$address[1] = "Beaconside";

$address[2] = "Stafford";

$postcode = "ST18 0DG";

$telephone = "01785 353419";

?>

Add HTML to your document to create a contact page using PHP to output the values of the variables created above. Make use of any HTML element that you feel is appropriate.

Your output should appear similar to the following:



## Exercise Two – Simple Form Processing

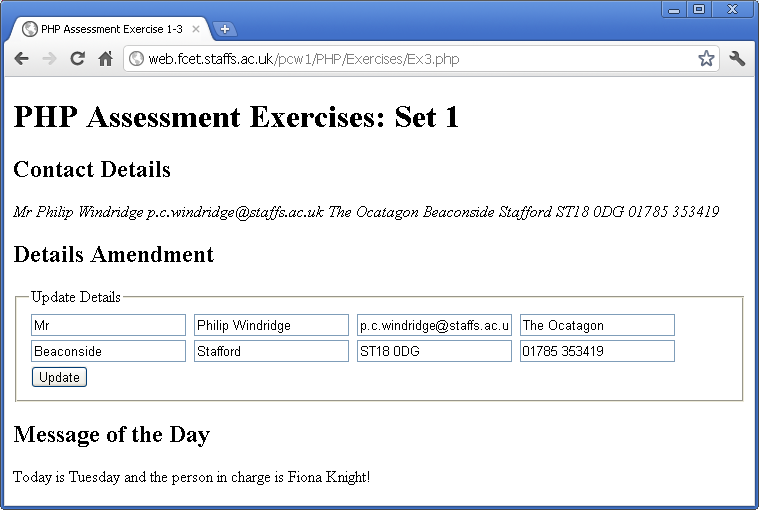


Create an HTML form in your webpage and amend/add PHP code that will allow a user to edit the variables given in Exercise One.

The user should be able to change the content of the form element and have those changes reflected in the variable’s output when the form is submitted. Make use of any HMTL element that you feel is appropriate.

Your output should appear similar to that shown above.

## Exercise Three – Built in functions



Add a ‘person in charge for the day’ message to appear which changes depending on what day of the week it is. For each day the person in charge will be as follows:

Monday = Russell Campion

Tuesday = Fiona Knight

Wednesday = Philip Windridge

Thursday = Fiona Knight

Friday = Russell Campion

Saturday/Sunday = Nobody

This information is coded into the following array. You should make use of this array ‘as is’ within your PHP code:

$personInCharge = array('Nobody',

'Russell Campion',

'Fiona Knight',

'Philip Windridge',

'Fiona Knight',

'Russell Campion',

'Nobody');

You should also make use of the date() function in PHP.

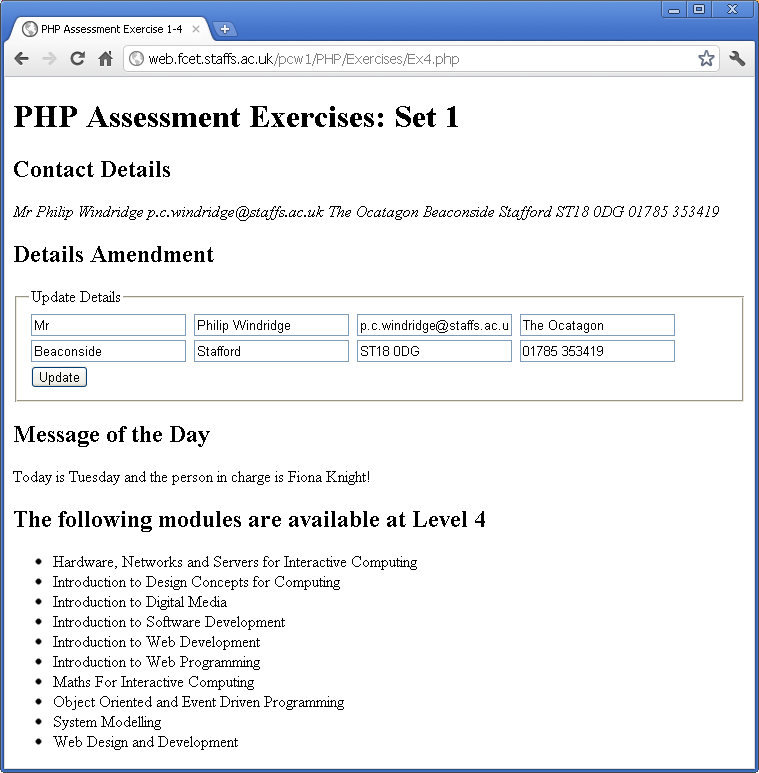
The message that your PHP outputs will take the form:

“Today is [dayOfWeek] and the person in charge is [personInCharge]!”

The messages should be selected using an appropriate conditional statement and whatever other PHP code that you feel is appropriate.

Your output should appear similar to that shown above.

## Exercise Four – Sorting an Array



Incorporate the following array into your code:

$moduleNames = array('Web Design and Development',

'Introduction to Web Programming',

'Introduction to Web Development',

'Introduction to Software Development',

'Object Oriented and Event Driven Programming',

'Hardware, Networks and Servers,

'Maths For Interactive Computing',

'System Modelling',

'Introduction to Design Concepts for Computing',

'Introduction to Digital Media');

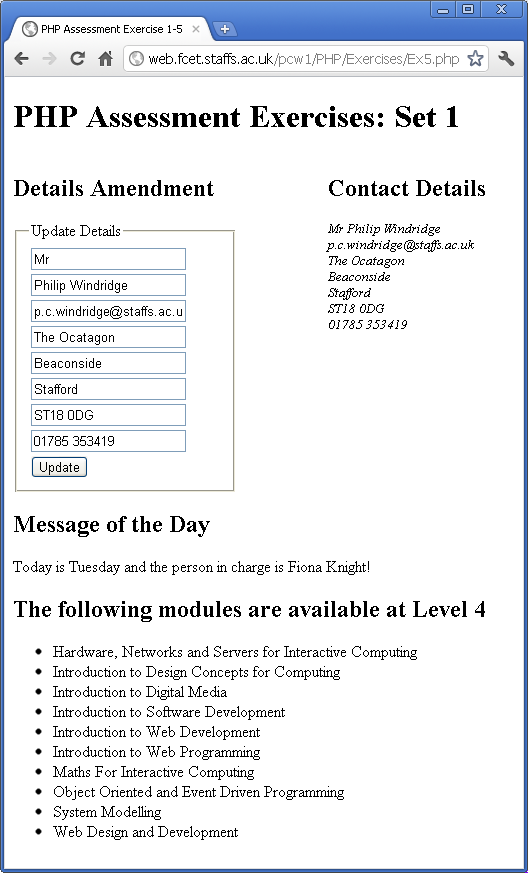
Output the array, using suitable HTML elements, as an alphabetically sorted list.

You can use any method you wish to sort the array. However, it is recommended that you consider the following function:

*sort()* – <http://www.php.net/sort>

Your output should appear similar to that shown above.

## Exercise Five – Applying Style



Using Cascading Style Sheets (CSS), style your web page so that the output is similar to the screenshot above.

In doing so you may add structure to your HTML using HTML elements such as <div> but removing style sheets altogether should result in the output appearing similar to that shown for Exercise 4.

# Assessment Criteria

## Contribution of Parts 1, 2 and 3 to the Coursework

Assessment Part 1, Assessment Part 2 and Assessment Part 3 will be handed out between weeks 2 and 4 inclusive. The three parts constitute 40% of the overall mark for the coursework. The coursework constitutes 100% of the marks available for the module.

Of the 40% contributed by Parts 1, 2 and 3 the individual allocation is as follows:

Part 1: 10% (this part)

Part 2: 14%

Part 3: 16%

**Total: 40%**

## Marks Allocation for Assessment Part 1

### Functionality marks (worth 100% of the marks available for Part 1)

2=Functionality completed, 1=Good attempt, 0=Poor attempt/No attempt

|  |  |  |
| --- | --- | --- |
|  | **Functionality** | **Marks (out of)** |
| Exercise 1 | Create PHP file | 2 |
| Place code block at beginning of file | 2 |
| Add HTML (semantic/standards compliant – validation proof required) | 2 |
| Output variable values | 2 |
| Exercise 2 | HTML form | 2 |
| Output values in form fields | 2 |
| Update values | 2 |
| No ‘notice’ warning | 2 |
| Exercise 3 | Add array of names | 2 |
| Select person according to day | 2 |
| Identify day using the date function | 2 |
| Output correct name for the day using correct message | 2 |
| Exercise 4 | Add array of modules | 2 |
| Sort modules alphabetically | 2 |
| Output module values | 2 |
| Use HTML list to output | 2 |
| Exercise 5 | Correct use of CSS | 2 |
| Web page displays correctly | 2 |
| Correct flow when stylesheets removed | 2 |
| CSS Validated – proof required | 2 |