

## Lab 6 – Arrays

Reference: Chapter 5 of the “PHP Programming with MySQL” textbook.

PHP Arrays <http://au.php.net/manual/en/book.array.php>

### Aims:

- To be able to use various array techniques and file functions to create another ‘guest book’ web application, similar to that created in Lab05, and similar to the example in the Lecture Notes, and check whether the user has previously signed the ‘guest book’ before adding the user details to the text file.

### Getting Started:

Create a new folder ‘lab06’ under the unit folder on the mercury server

~/cos30020/www/htdocs folder on mercury.

Save today’s work in this lab06 folder.

All Web pages must be validated

You could also create and link an external stylesheet, to the pages, and this should be valid CSS.

## Task 1: Understanding file and array functions (9 marks)

### Step 1:

Create a file **shoppingsave.php** that will receive the input data from **shoppingform.php** from Step 2 via POST method, saves the item and quantity into a text file called “**shop.txt**” in the “**data**” directory, and then output all items in the text file arranged alphabetically by the item name. The item and quantity is to be delimited by comma when saved into the text file. If the item already exists, the list is not updated. For this exercise, no input validation aside from **isset()** is performed.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="description" content="Web application development" />
  <meta name="keywords" content="PHP" />
  <meta name="author" content="Your Name" />
  <title>TITLE</title>
</head>
<body>
<h1>Web Programming - Lab06</h1>
<?php // read the comments for hints on how to answer each item
  if ( (1) ) { // check if both form data exists
    $item = $_POST[" (2) "]; // obtain the form item data
    $qty = $_POST[" (3) "]; // obtain the form quantity data
    $filename = "../..data/shop.txt"; // assumes php file is inside lab06

    $alldata = array(); // create an empty array
    if ( (4) ) { // check if file exists for reading
      $itemdata = (5); // create an empty array
      $handle = fopen( (6), "r"); // open the file in read mode
      while (! feof ( (7) )) { // loop while not end of file
        $onedata = fgets($handle); // read a line from the text file
        if ($onedata != "") { // ignore blank lines
          $data = (8); // explode string to array
          $alldata [] = $data; // create an array element
          $itemdata[] = $data [0]; // create a string element
        }
      }
    }
    fclose ( (9) ); // close the text file
    $newdata = !( (10) ($item, $itemdata)); // check if item exists in array
```

```
} else {
    $newdata = true;    // file does not exists, thus it must be a new data
}
if ($newdata) {
    $handle = fopen( (11), "a");    // open the file in append mode
    $data = $item . "," . $qty . "\n"; // concatenate item and qty delimited
                                        // by comma
    fputs( (12), $data);           // write string to text file
    fclose ( (13) );               // close the text file

    $alldata [] = array($item, $qty); // add data to array

    echo "<p>Shopping item added</p>";
} else {
    echo "<p>Shopping item already exists</p>";
}

(14) ($alldata);                // sort array elements

echo "<p>Shopping List</p>";
foreach ( (15) as (16) ) {        // loop using for each
    echo "<p>", $data [0], " -- ", $data[1], "</p>";
}
} else {                            // no input
    echo "<p>Please enter item and quantity in the input form.</p>";
}
?>
</body>
</html>
```

## Step 2:

Create a file **shoppingform.php** that contains a form with a two text boxes that allows a user to enter the item name to be purchases and a number representing the quantity, and submit it to **shoppingsave.php**.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8" />
    <meta name="description" content="Web application development" />
    <meta name="keywords" content="PHP" />
    <meta name="author" content="Your Name" />
    <title>TITLE</title>
</head>
<body>
    <h1>Web Programming Form - Lab06</h1>
    <form action = (17) method = (18) >
        (19-20)
        _____
        _____
        _____
        _____
    </form>
</body>
</html>
```

## Task 2: Updating the Guest Book System (6 marks)

### Step 1:

Create a file **guestbookform.php** as shown below, that enables visitors to enter their name and email address. Form action is to **guestbooksave.php** with method **post**.  
(You could copy and modify your work from the previous lab)

The “View Guest Book” link is to **guestbookshow.php**.

### Lab06 Task 2 - Guestbook

Enter your details to sign our guest book

Name: 
E-mail:

[View Guest Book](#)

### Step 2:

Create a file **guestbooksave.php** with a script that checks the data entered, and then stores the visitor's name and email address in a text file **guestbook.txt**.

Using array techniques that you learned in Module 6, include functionality that prevents the same name or email address from being stored twice.

**Hint:** Read the file into an array, then use `in_array()` to check that the name or email does not already exist.

**Note:** The `visitor.txt` file should be in a 'lab06' directory under the 'data' directory on mercury. Make the 'lab06' directory if it does not exist. Use the relative file path '`../../data/lab06/guestbook.txt`'

<h3>Lab06 Task 2 - Guestbook</h3> <h4>Sign Guestbook</h4> <hr/> <p>Thank you for signing our guest book:</p> <p><b>Name:</b> Freddy "marvel" Bloggs <b>E-mail:</b> fmarvel@bloggs.com.au</p> <hr/> <p><a href="#">Add Another Visitor</a> <a href="#">View Guest Book</a></p>	<h3>Lab06 Task 2 - Guestbook</h3> <h4>Sign Guestbook</h4> <hr/> <p><b>You have already signed the guest book!</b></p> <hr/> <p><a href="#">Add Another Visitor</a> <a href="#">View Guest Book</a></p>
<h3>Lab06 Task 2 - Guestbook</h3> <h4>Sign Guestbook</h4> <hr/> <p><b>You must enter your name and email address!</b> <b>Use the Browser's 'Go Back' button to return to the Guestbook form.</b></p> <hr/> <p><a href="#">Add Another Visitor</a> <a href="#">View Guest Book</a></p>	

### Step 3:

Create a file **guestbookshow.php** with a script to display a webpage that shows all the visitors that have signed the Guest Book in a table. Use the array techniques that you learned in Module 6 to read the file into an array, sort the array by Name, and display the Guestbook.

## Lab06 Task 2 - Guestbook

### View Guestbook

Number	Name	Email
1	Freddy "marvel" Bloggs	fmarvel@bloggs.com.au
2	Little Johnny Five	jfive@factory.co.nz

[Add Another Visitor](#)

**Challenge:** Validating email input in `guestbooksave.php`

Use array functions, string functions, or regular expressions, to check that the entered email values are in an 'email format'

**Hint:**

```
if (preg_match($regexp, $email)) {  
    echo "<p>Email address is valid.</p>";  
} else {  
    echo "<p>Email address is not valid.</p>";  
}
```