CNT 4714: Enterprise Computing Spring 2012

Introduction to PHP – Part 3 - Arrays

Instructor : Dr. Mark Llewellyn
markl@cs.ucf.edu
HEC 236, 407-823-2790
http://www.cs.ucf.edu/courses/cnt4714/spr2012

Department of Electrical Engineering and Computer Science
Computer Science Division
University of Central Florida

Arrays In PHP

- Most of our PHP examples to this point have involved scalar variables (we did see a couple of example in the first section of notes that made use of one of PHP's global associative arrays).
- Scalar variables can only hold a single value at a time. For example, a variable `$color` could hold only a single value such as `red`, at any point in time. The variable could not be used to hold more than one color.
- Arrays are special types of variables that enable you to store as many values as you want.

Note: Although you can technically make an array as large as you'd like, some built-in array handling functions in PHP have an upper limit of 100,000 values. If you are storing more data that this in your arrays and you need to use one of these functions, you will either need to write your own function or split the data into multiple arrays.

Arrays In PHP

- Arrays are indexed, which means that each entry in the array, called an element, is made up of a key and a value.
- The key is the index position, beginning with 0 and increasing incrementally by 1 with each new element in the array.
- The value is whatever value you associate with that position – a string, an integer, or whatever you want.
- In PHP you can think of an array as a filing cabinet and each key/value pair as a file folder. The key is the label written on the tab of the folder, and the value is what is inside. What's inside each folder can vary from folder to folder.

Creating Arrays In PHP

- You can create an array using either the `array()` function or the array operator `[]`.
- The `array()` function is usually used when you want to create a new array and populate it with more than one element, all at the same time.
- The array operator is more often used when you want to create a new array with just one element at the outset or when you want to add to an existing array element.
- The examples on the following couple of pages illustrate creating an array in PHP using these two techniques.

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\array definition V1.php - Notepad++

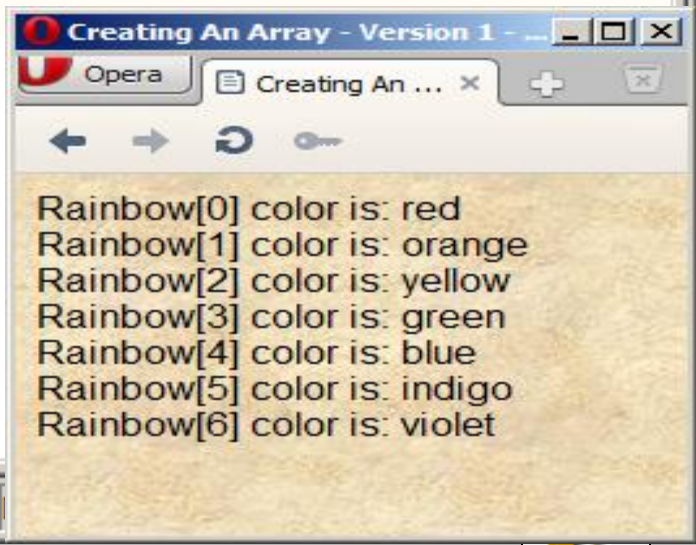
File Edit Search View Encoding Language Settings Macro Run Plugins Window ? X

PS-Part5-p15-signed.ps1 | Project9.ps1 | Get-StoppedService.ps1 | array definition V1.php

```
1  <html>
2  <head>
3  <title>Creating An Array - Version 1</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $rainbow = array("red","orange","yellow","green","blue","indigo","violet");
9      for ($i=0; $i<=6; $i++) {
10         echo 'Rainbow['. $i.'] color is: '. $rainbow[$i]."<br />";
11     }
12 ?>
13 </body>
14 </html>
```

PHP Hypert | length : 384 lines : 14 | Ln : 1 Col : 1 Sel : 0

This version uses the `array()` function to create the array.



C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\array definition V2.php - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Windo X

PS-Part5-p15-signed.ps1 | Project9.ps1 | Get-StoppedService.ps1 | array de

This version uses the array operator [ ] to create the array.
Note that no index values are specified, PHP will auto number for you

```
1  <html>
2  <head>
3  <title>Creating An Array - Version 2</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $rainbow[] = "red";
9      $rainbow[] = "orange";
10     $rainbow[] = "yellow";
11     $rainbow[] = "green";
12     $rainbow[] = "blue";
13     $rainbow[] = "indigo";
14     $rainbow[] = "violet";
15     for ($i=0; $i<=6; $i++) {
16         echo 'Rainbow['. $i.'] color is: '. $rainbow[$i]."<br />";
17     }
18 ?>
19 </body>
20 </html>
```

Creating An Array - Version 2 - ...

Opera | Creating An ...

Rainbow[0] color is: red
Rainbow[1] color is: orange
Rainbow[2] color is: yellow
Rainbow[3] color is: green
Rainbow[4] color is: blue
Rainbow[5] color is: indigo
Rainbow[6] color is: violet

PHP Hypert | length : 475 lines : 20 | Ln : 1 Col : 1 Sel : 0 | Dos\Windows | ANSI | INS

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\array definition V3.php - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window

Project9.ps1 | Get-StoppedService.ps1 | array definition V1.php | array definiti

This version also uses the array operator [ ] to create the array.
Note that index values are specified in this case.

```
1  <html>
2  <head>
3  <title>Creating An Array - Version 3</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.
7  <?php
8      $rainbow[0] = "red";
9      $rainbow[1] = "orange";
10     $rainbow[2] = "yellow";
11     $rainbow[3] = "green";
12     $rainbow[4] = "blue";
13     $rainbow[5] = "indigo";
14     $rainbow[6] = "violet";
15     for ($i=0; $i<=6; $i++) {
16         echo 'Rainbow['. $i.'] color is: '. $rainbow[$i]."<br />";
17     }
18 ?>
19 </body>
20 </html>
```

Creating An Array - Version 3 - Op...

Opera | Creating An Arra...

Rainbow[0] color is: red
Rainbow[1] color is: orange
Rainbow[2] color is: yellow
Rainbow[3] color is: green
Rainbow[4] color is: blue
Rainbow[5] color is: indigo
Rainbow[6] color is: violet

PHP Hypert | length : 482 lines : 20 | Ln : 1 Col : 1 Sel : 0 | Dos\Windows | ANSI | INS

Creating Arrays In PHP

- As shown in the example on page 6, PHP can automatically index the array for you when you use the [ ] operator to create the array.
- This is useful in that it eliminates the possibility that you might misnumber the elements. The example on the next page illustrates what happens if you misnumber the elements in an array.

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\array definition V4 misnumbering.php - ...

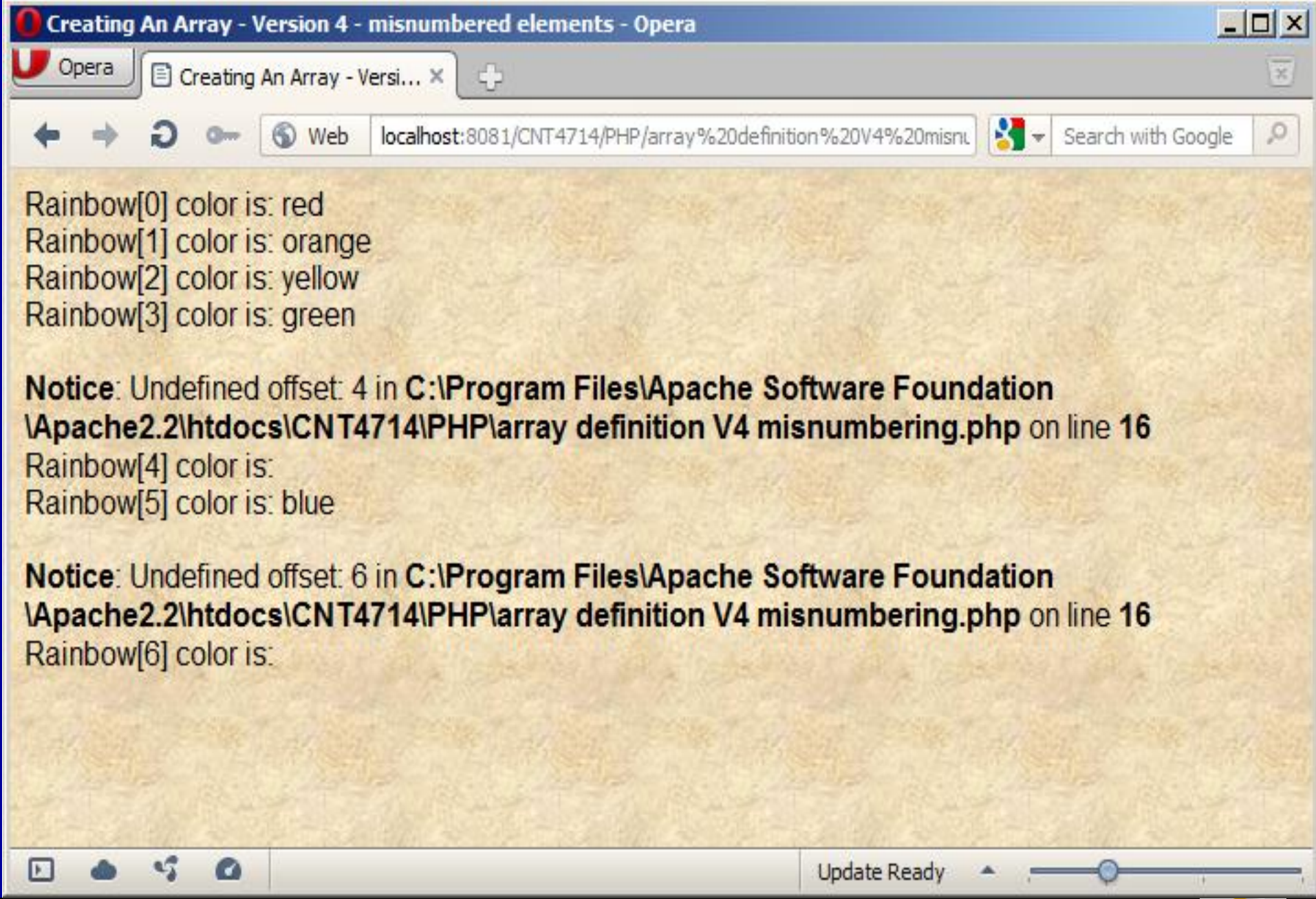
File Edit Search View Encoding Language Settings Macro Run Plugins Window ? X

array definition V1.php | array definition V2.php | array definition V3.php | array definition V4 misnumbering.php

```
1  <html>
2  <head>
3  <title>Creating An Array - Version 4 - misnumbered elements</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $rainbow[0] = "red";
9      $rainbow[1] = "orange";
10     $rainbow[2] = "yellow";
11     $rainbow[3] = "green";
12     $rainbow[5] = "blue";
13     $rainbow[7] = "indigo";
14     $rainbow[8] = "violet";
15     for ($i=0; $i<=6; $i++) {
16         echo 'Rainbow['. $i.'] color is: '. $rainbow[$i]."<br />";
17     }
18 ?>
19 </body>
20 </html>
```

Misnumbering starts here with no element 4 defined and then 6 too is missed.

PHP Hypert | length : 505 lines : 20 | Ln : 1 Col : 1 Sel : 0 | Dos\Windows | ANSI | INS


Creating An Array - Version 4 - misnumbered elements - Opera
Opera
Creating An Array - Versi...
Web
localhost:8081/CNT4714/PHP/array%20definition%20V4%20misnu
Search with Google
Rainbow[0] color is: red
Rainbow[1] color is: orange
Rainbow[2] color is: yellow
Rainbow[3] color is: green
Notice: Undefined offset: 4 in C:\Program Files\Apache Software Foundation \Apache2.2\htdocs\CNT4714\PHP\array definition V4 misnumbering.php on line 16
Rainbow[4] color is:
Rainbow[5] color is: blue
Notice: Undefined offset: 6 in C:\Program Files\Apache Software Foundation \Apache2.2\htdocs\CNT4714\PHP\array definition V4 misnumbering.php on line 16
Rainbow[6] color is:
Update Ready

Creating Associative Arrays In PHP

- The arrays we've seen so far have been numerically indexed, meaning that they use an integer index position as the key.
- Associative arrays utilize actual named keys. In PHP, the named keys of an associative array are character strings rather than numerical values. The string value is used to look up or provide a cross-reference to the data value.
- The following example creates an associative array named `$instructor` with three elements.

```
$instructor["CNT 4714"] = "Llewellyn";

$instructor["CIS 3003"] = "Eisler";

$instructor["CIS 3360"] = "Guha";
```

Creating Associative Arrays In PHP

- The same array could also be created using the `array()` function instead of the array operator [ ]. This is shown below:

```
$instructor = array ("CNT 4714" => "Llewellyn",
"CIS 3003" => "Eisler", "CIS 3360" => "Guha");
```

- When using the `array()` function, items are assigned in index/value pairs using the => operator.

- When you want to access an item in an associative array, a syntax similar to that used with sequential (numerically indexed) arrays is employed, however, a string value or variable is used for the index.

Creating Associative Arrays In PHP

- Suppose you wanted to retrieve the instructor for CIS 4004. The following expression would achieve this:

```
$teacher = $instructor["CNT 4714"];
```

- The variable `$teacher` would be assigned the data value associated with "CNT 4714" which would be "Llewellyn".

Note: You might be tempted to do the following with an associative array, where you are trying to determine which course is taught by the instructor named "Llewellyn":

```
$course = $instructor["Llewellyn"];
```

Don't do this! An associative array can fetch data values only via the keys and not the values associated with the keys. Therefore, it cannot find and entry in the array with an index value of "Llewellyn" and will return nothing and the value of $course will be undefined. The example on the following page illustrates this.

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\associative array - incorrect version.ph...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ? X

array definition V2.php | array definition V3.php | array definition V4 misnumbering.php | associative array - incorrect version.php

```
1  <html>
2  <head>
3  <title>Creating An Associative Array - Incorrect Version</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $instructor = array( "CNT 4714" => "Llewellyn",
9                           "CIS 3003" => "Eisler",
10                          "CIS 3360" => "Guha" );
11     echo 'The instructor of CNT 4714 is: '. $instructor["CNT 4714"]. "<br />";
12     echo 'The course taught by Eisler is: '. $instructor["Eisler"]. "<br />";
13
14 ?>
15 </body>
16 </html>
```

Incorrect Version

PHP Hypert | length : 515 lines : 16

Creating An Associative Array - Incorrect Version - Opera

Opera | Creating An Associative ... | Web | localhost:8081/CNT4714/PHP/associative%20array%20-%20incor | Search with Google

The instructor of CNT 4714 is: Llewellyn

Notice: Undefined index: Eisler in **C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\associative array - incorrect version.php** on line **12**
The course taught by Eisler is:

Update Ready

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\associative array - correct version.php ...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

array definition V4 misnumbering.php | associative array - incorrect version.php | associative array - correct version.php

Correct Version

```
1  <html>
2  <head>
3  <title>Creating An Associative Array - Correct Version</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $instructor = array( "CNT 4714" => "Llewellyn",
9                           "CIS 3003" => "Eisler",
10                          "CIS 3360" => "Guha" );
11     echo 'The instructor of CNT 4714 is: '. $instructor["CNT 4714"]. "<br />";
12     echo 'The instructor of CIS 3003 is: '. $instructor["CIS 3003"]. "<br />";
13
14 ?>
15 </body>
16 </html>
```

PHP Hypert length : 514 lines : 16 INS

Creating An Associative Array - Correct Version - Opera

Opera | Creating An Associative ... | Web | localhost:8081/CNT4714/PHP/associative% | Search with

The instructor of CNT 4714 is: Llewellyn
The instructor of CIS 3003 is: Eisler

Update Ready

Using Associative Arrays In PHP

- A common iterative statement used with both sequential and associative arrays is the `foreach` statement.
- The general syntax of the `foreach` statement is:

```
foreach ( arrayname as variable ) {

    . . . Statements to repeat

}
```

- The first variable inside the parentheses is the variable name representing the array and the second variable is automatically set to the next array item at each iteration of the loop. An example using a sequential array is shown on the next page and one with an associative array on the following page.

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\foreach

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

associative array - incorrect version.php | associative array - correct v

Zoom In (Ctrl+Mouse Wheel U

```
1  <html>
2  <head>
3  <title>Foreach statement</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $languages = array( "PHP", "C", "Java", "Python", "Ruby", "Pascal", "Fortran");
9      echo "<b> A list of computer languages </b> <br /> <br />";
10     foreach ($languages as $item) {
11         echo $item . "<br />";
12     }
13 ?>
14 </body>
15 </html>
```

Foreach statement - Opera

Opera | Foreach statement

Web | localhost:8081/C

A list of computer languages

PHP
C
Java
Python
Ruby
Pascal
Fortran

Update Ready

PHP Hypert | length : 408 lines : 15 | Ln : 9 Col : 58 Sel : 0 | Dos\Windows | ANSI | INS

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\foreach statement - associative array.p...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ? X

associative array - correct version.php | foreach statement.php | foreach statement - associative array.php

```
1  <html>
2  <head>
3  <title>Foreach statement with an associative array</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $inventory = array( "hard drives" => 10, "printers" => 4, "monitors" => 23);
9      echo "Current Inventory <br /> <br />";
10     foreach ($inventory as $index => $item) {
11         echo $index . ' = ' . $item . "<br />";
12     }
13 ?>
14 </body>
15 </html>
```

PHP Hypert | length : 438 lines : 15 | Ln : 1 Col : 1 Sel : 0

Foreach statement with an asso...

Opera | Foreach state...

Current Inventory

hard drives = 10
printers = 4
monitors = 23

Using Associative Arrays In PHP

- Changing values, adding elements, deleting elements, and verifying an element are all among the common operations that you'll need to perform on an associative array.
- Changing an existing value is done through simple assignment. For example, to update the number of monitors in the previous example from 23 to 5, the following statement would be used: `$inventory["monitors"] = 5;`
- To add a new element to an associative array, use the array operator [] as in: `$inventory["keyboards"] = 12;`
- Deleting an element from an associative array is done using the `unset()` function.

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\associative array operations.php - Note...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ? X

associative array operations.php

```
2   <head>
3   <title>Some operations on an associative array</title>
4   </head>
5   <body  style = "font-family: arial, sans-serif;
6         background-color: #856363"  background=image1.jpg>
7   <?php
8       $inventory = array( "hard drives" => 10, "printers" => 4, "m
9       echo "<b><u>Current Inventory</u></b> <br /> <br />";
10      foreach ($inventory as $index => $item) {
11          echo $index . ' = ' . $item . "<br />";
12      }
13      $inventory["monitors"] = 5;
14      $inventory["keyboards"] = 12;
15      echo "<br /><b><u> Current Inventory</u></b> <br /> <br />";
16      foreach ($inventory as $index => $item) {
17          echo $index . ' = ' . $item . "<br />";
18      }
19      unset($inventory["printers"]);
20      echo "<br /><b><u> Current Inventory</u></b> <br /> <br />";
21      foreach ($inventory as $index => $item) {
22          echo $index . ' = ' . $item . "<br />";
23      }
24  ?>
25  </body>
```

PHP Hypert | length : 857 lines : 26 | Ln : 20 Col : 17 Sel : 0 | Dos\Windows

Some operations on an as...

Opera | Some ...

Current Inventory

hard drives = 10
printers = 4
monitors = 23

Current Inventory

hard drives = 10
printers = 4
monitors = 5
keyboards = 12

Current Inventory

hard drives = 10
monitors = 5
keyboards = 12

Using Associative Arrays In PHP

- To verify if a particular index exists in an associative array, use the `isset()` function.
- The `isset()` function returns true if index passed as an argument appears in the associative array and false otherwise.
- The example on the following page illustrates using the `isset()` function.

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\using the isset() function.php - Notepad...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ? X

associative array operations.php | using the isset() function.php

```
1  <html>
2  <head>
3  <title>Using the isset() function on an associative array</title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $inventory = array( "hard drives" => 10, "printers" => 4, "monitors" => 23);
9      echo "<b><u>Current Inventory</u></b> <br /> <br />";
10     foreach ($inventory as $index => $item) {
11         echo $index . ' = ' . $item . "<br />";
12     }
13     echo "<br />";
14     if ( isset($inventory["monitors"]) ) {
15         echo "monitors is in the array. <br />";
16     }else { echo "monitors is not in the array. <br />";
17     }
18     if ( isset($inventory["keyboards"]) ) {
19         echo "keyboards is in the array. <br />";
20     }else { echo "keyboards is not in the array. <br />";
21     }
22 ?>
23 </body>
24 </html>
```

PHP Hypert | length : 773 lines : 24 | Ln : 24 Col : 8 Sel : 0 | Dos\Win

Using the isset() function on an as...

Opera | Using the isset() ... ×

Current Inventory

hard drives = 10
printers = 4
monitors = 23

monitors is in the array.
keyboards is not in the array.

Using Associative Arrays In PHP

- As with many things in PHP, associative array indices are case-sensitive. Thus, in the previous example, if the call to the `isset()` function were passed the parameter "Monitors" instead of "monitors" it would return false instead of true.
- See next page.

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\using the isset() function - case sensitiv...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ? X

associative array operations.php | using the isset() function.php | using the isset() function - case sensitivity.php

```
1  <html>
2  <head>
3  <title>Using the isset() function on an associative array<
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7  <?php
8      $inventory = array( "hard drives" => 10, "printers" =>
9      echo "<b><u>Current Inventory</u></b> <br /> <br />";
10     foreach ($inventory as $index => $item) {
11         echo $index . ' = ' . $item . "<br />";
12     }
13     echo "<br />";
14     //associative array indices are case-sensitive - monitors is in array
15     //Monitors is not
16     if ( isset($inventory["Monitors"]) ) {
17         echo "monitors is in the array. <br />";
18     }else { echo "monitors is not in the array. <br />";
19     }
20     if ( isset($inventory["keyboards"]) ) {
21         echo "keyboards is in the array. <br />";
22     }else { echo "keyboards is not in the array. <br />";
23     }
24 ?>
```

Using the isset() function on an as...

Opera | Using the isset() ...

Current Inventory

hard drives = 10
printers = 4
monitors = 23

monitors is not in the array.
keyboards is not in the array.

PHP Hypert | length : 865 lines : 26 | Ln : 24 Col : 3 Sel : 0 | Dos\Windows | ANSI | INS

Sorting Associative Arrays In PHP

- PHP has a special set of functions for sorting associative arrays.
- The `asort()` function sorts an associative array and maintains the relationship between the indices and the values. The sort is based upon the values in the associative array passed as an argument to the function. The sort order is ascending based on the value. The `arsort()` function sorts in descending order based on value.
- The `ksort()` function is similar to the `asort()` function but it sorts an associative array using the indices (in ascending order) as the sort field. The `krsort()` function sorts in descending order using the indices.
- These various sort functions are shown on the next few pages.

Using `asort()`

```php
<html>
<head>
    <title> Using the asort() function </title>
</head>
<body  style = "font-family: arial, sans-serif;
      background-color: #856363"  background=image1.
<?php
    $cities['Boston'] = 1302;
    $cities['Dallas'] = 1093;
    $cities['Toronto'] = 1282;
    $cities['Chicago'] = 1154;
    $cities['Los Angeles'] = 2512;
    $cities['San Francisco'] = 2825;
    $cities['Washington']= 853;
    $cities['Miami'] = 235;
    $cities['Pittsburgh'] = 972;
    $cities['Tampa'] = 85;

    asort($cities);
    print("Cities sorted by distance from Orlando. <br /><br />");
    foreach ($cities as $index => $value) {
        print (" $index = $value <br />");
    }
?>
```

Using the asort() function - Opera

Cities sorted by distance from Orlando.

Tampa = 85
Miami = 235
Washington = 853
Pittsburgh = 972
Dallas = 1093
Chicago = 1154
Toronto = 1282
Boston = 1302
Los Angeles = 2512
San Francisco = 2825

Using `arsort()`

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\using the arsort() function.php - Notepa...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

using the arsort() function.php

```
1  <html>
2  <head>
3      <title> Using the arsort() function </title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.
7  <?php
8      $cities['Boston'] = 1302;
9      $cities['Dallas'] = 1093;
10     $cities['Toronto'] = 1282;
11     $cities['Chicago'] = 1154;
12     $cities['Los Angeles'] = 2512;
13     $cities['San Francisco'] = 2825;
14     $cities['Washington']= 853;
15     $cities['Miami'] = 235;
16     $cities['Pittsburgh'] = 972;
17     $cities['Tampa'] = 85;
18
19     arsort($cities);
20     print("<b><u>Cities sorted by distance from Orlando.</u></b> <br /><br />");
21     foreach ($cities as $index => $value) {
22         print (" $index = $value <br />");
23     }
24 ?>
```

Using the arsort() function - Opera

Opera | Using the arsort() function | Web | localhost:8081/C

Cities sorted by distance from Orlando.

San Francisco = 2825
Los Angeles = 2512
Boston = 1302
Toronto = 1282
Chicago = 1154
Dallas = 1093
Pittsburgh = 972
Washington = 853
Miami = 235
Tampa = 85

Update Ready

PHP Hypert | length : 666 lines : 28 | Ln : 1 Col : 7 Sel : 0 | UNIX | ANSI | INS

Using `ksort()`

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\using the ksort() function.php - Notepa...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

using the arsort() function.php | using the ksort() function.php

```
1  <html>
2  <head>
3      <title> Using the ksort() function </title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.
7  <?php
8      $cities['Boston'] = 1302;
9      $cities['Dallas'] = 1093;
10     $cities['Toronto'] = 1282;
11     $cities['Chicago'] = 1154;
12     $cities['Los Angeles'] = 2512;
13     $cities['San Francisco'] = 2825;
14     $cities['Washington']= 853;
15     $cities['Miami'] = 235;
16     $cities['Pittsburgh'] = 972;
17     $cities['Tampa'] = 85;
18
19     ksort($cities);
20     print("<b><u>Cities sorted by distance from Orlando.</u></b> <br /><br />");
21     foreach ($cities as $index => $value) {
22         print (" $index = $value <br />");
23     }
24 ?>
```

Using the ksort() function - Opera

Opera | Using the ksort() function | Web | localhost:8081/C

Cities sorted by distance from Orlando.

Boston = 1302
Chicago = 1154
Dallas = 1093
Los Angeles = 2512
Miami = 235
Pittsburgh = 972
San Francisco = 2825
Tampa = 85
Toronto = 1282
Washington = 853

Update Ready

PHP Hypert | length : 664 lines : 28 | Ln : 1 Col : 7 Sel : 0 | UNIX | ANSI | INS

Using `krsort()`

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\using the krsort() function.php - Notepa...

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

using the arsort() function.php | using the ksort() function.php | using the krsort() function.php

```
1  <html>
2  <head>
3      <title> Using the krsort() function </title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.
7  <?php
8      $cities['Boston'] = 1302;
9      $cities['Dallas'] = 1093;
10     $cities['Toronto'] = 1282;
11     $cities['Chicago'] = 1154;
12     $cities['Los Angeles'] = 2512;
13     $cities['San Francisco'] = 2825;
14     $cities['Washington']= 853;
15     $cities['Miami'] = 235;
16     $cities['Pittsburgh'] = 972;
17     $cities['Tampa'] = 85;
18
19     krsort($cities);
20     print("<b><u>Cities sorted by distance from Orlando.</u></b> <br /><br />");
21     foreach ($cities as $index => $value) {
22         print (" $index = $value <br />");
23     }
24 ?>
```

PHP Hypert | length : 666 lines : 28 | Ln : 20 Col : 65 Sel : 0 | UNIX | ANSI | INS

Using the krsort() function - Opera

Opera | Using the krsort() function | Web | localhost:8081/C

Cities sorted by distance from Orlando.

Washington = 853
Toronto = 1282
Tampa = 85
San Francisco = 2825
Pittsburgh = 972
Miami = 235
Los Angeles = 2512
Dallas = 1093
Chicago = 1154
Boston = 1302

Update Ready

Using Multidimensional Arrays In PHP

- Some data are best represented by creating a list of lists (a multidimensional array).
- Consider the following table listing the inventory for a hardware store:

Part Number	Part Name	Count	Price
AC1000	Hammer	122	28.50
AC1001	Wrench	25	14.00
AC1002	Saw	18	25.00
AC1003	Screwdriver	34	4.50

- The example on the next page represents this data in a two-dimensional associative array.

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\drive_inventory.php - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

drive_inventory.php | show_inventory.php

```
1  <html>
2  <head>
3      <title> Hardware Inventory </title>
4  </head>
5  <body  style = "font-family: arial, sans-serif;
6        background-color: #856363"  background=image1.jpg>
7
8      <form action="show_inventory.php" method = "post">
9      <font size=4 color ="blue"><b><u>Mark's Hardware Inventory Information</u></b></font><br />
10     <br /><br />Select part number for more information </font> <br /><br />
11     <?php
12         $Inventory =  array( 'AC1000', 'AC1001', 'AC1002', 'AC1003' );
13         foreach ( $Inventory as $item ) {
14                 print "<input type=radio name=\"id\" value=$item > $item ";
15         }
16         print '<br /><br /><input type=submit value="Submit">   ';
17         print '<input type=reset value="Erase ">';
18         print '</form> </body></html>';
19     ?>
20 </body>
21 </html>
22
```

Front-end
Provides a set of radio buttons for user to select the part they'd like to see more information about.

PHP Hypertext Preprocessor | length : 736 lines : 22 | Ln : 21 Col : 8 Sel : 0 | UNIX | ANSI | INS

C:\Program Files\Apache Software Foundation\Apache2.2\htdocs\CNT4714\PHP\show_inventory.php - Notepad++

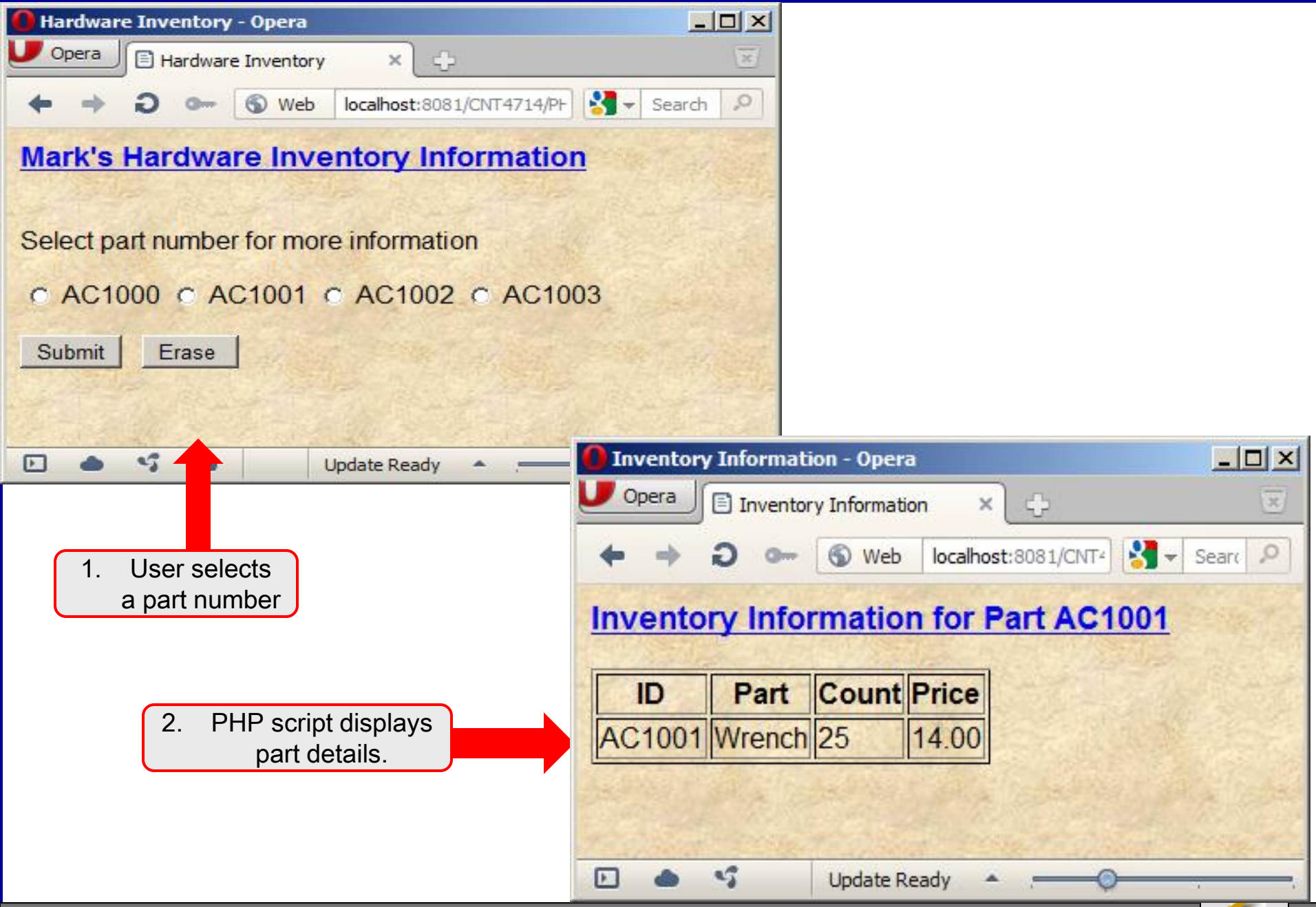
File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

drive_inventory.php | show_inventory.php

```
<?php
 $inventory = array (
        'AC1000'=>array('Part'=>'Hammer','Count'=>122,
                                        'Price'=> 28.50 ),
        'AC1001' => array('Part' =>'Wrench','Count' =>25,
                                          'Price'=>14.00 ),
         'AC1002'=>array('Part' =>'Handsaw','Count' =>18,
                                        'Price'=>25.00 ),
         'AC1003'=>array('Part' =>'Screwdrivers','Count'=>34,
                                        'Price'=>4.50)
        );
      $id = $_POST["id"];
     if (isset($inventory[$id])){
        print '<font size=4 color="blue"> ';
        print "<b><u>Inventory Information for Part $id </u></b></font><br /><br />";
        print '<table border=1> <th> ID <th> Part <th> Count <th> Price ';
        print "<tr> <td> $id </td>";
            print "<td> {$inventory[$id]['Part']} </td>";
            print "<td> {$inventory[$id]['Count']} </td>";
            print("<td>");
              printf( "%4.2f",$inventory[$id]['Price'] );
             print("</td></tr>");
         } else {
            print "Illegal part ID = $id ";
         }
?>
</body>
```

PHP Hypertext Preprocessor | length : 1109 | lines : 36 | INS

PHP script to find the correct entry in the associative array based on the user's selection, then display the associative array entries for that item.


Hardware Inventory - Opera
Opera
Hardware Inventory
Web
localhost:8081/CNT4714/PH
Search
Mark's Hardware Inventory Information
Select part number for more information
AC1000 AC1001 AC1002 AC1003
Submit
Erase
Update Ready
1. User selects a part number
2. PHP script displays part details.
Inventory Information - Opera
Opera
Inventory Information
Web
localhost:8081/CNT4
Searc
Inventory Information for Part AC1001
ID
Part
Count
Price
AC1001
Wrench
25
14.00
Update Ready