Chapter 11

How to create and use arrays

Objectives

Applied

1. Use any of the functions and techniques presented in this chapter as you use arrays, associative arrays, and arrays of arrays.

Objectives (continued)

Knowledge

1. Distinguish between an array and an associative array, including the difference in the way indexes are used.
2. Explain how gaps can be introduced into an array and how the gaps can be removed.
3. Describe the use of a for loop with an array and the use of a foreach loop with an associative array.
4. Distinguish between a queue and a stack.
5. Describe the use of the functions for creating arrays, working with queues and stacks, performing mathematical calculations, searching arrays, sorting arrays, and modifying arrays.
6. Distinguish between a regular array and an array of arrays, including the difference in the number of indexes that are used to access an element.

The syntax for creating an array

```
$array_name = array([value1[, value2] ...])
```

The syntax for referring to an element an array

```
$array_name[index];
```

How to create an array of names

With one statement

```
$names = array('Ted Lewis', 'Sue Jones', 'Ray Thomas');
```

With multiple statements

```
$names = array();         // create an empty array
$names[0] = 'Ted Lewis';  // set 3 values in the array
$names[1] = 'Sue Jones';
$names[2] = 'Ray Thomas';
```


How to create an array of discounts

With one statement

```
$discounts = array(0, 5, 10, 15);
```

With multiple statements

```
$discounts = array(); // create an empty array
$discounts[0] = 0;    // set 4 values in the array
$discounts[1] = 5;
$discounts[2] = 10;
$discounts[3] = 15;
```


How to use the print_r() function to view the contents of an array

```
print_r($names);
// Output: Array ( [0] => Ted Lewis [1] => Sue Jones
// [2] => Ray Thomas )
```

How to define a constant for an array (PHP 7.0 or later)

```
Define('MONTHS', array('Jan', 'Feb', 'Mar'));
```


Key terms

- array
- element
- index

The syntax for adding an element to an array

```
$array_name[] = $value;
```

How to add a value to the end of an array

```
$letters = array('a', 'b', 'c', 'd');  // a, b, c, d
$letters[] = 'e';                      // a, b, c, d, e
```

How to set a value at a specific index

```
$letters = array('a', 'b', 'c', 'd');  // a, b, c, d
$letters[0] = 'e';                     // e, b, c, d
$letters[3] = 'f';                     // e, b, c, f
$letters[5] = 'g';               // e, b, c, f, NULL, g
```

How to get values from an array

```
$letters = array('a', 'b', 'c', 'd'); // a, b, c, d
$letter1 = $letters[0];               // $letter1 is 'a'
$letter2 = $letters[1];               // $letter2 is 'b'
$letter4 = $letters[4];               // $letter4 is NULL
```


Functions for removing the values from elements

```
unset($var1[, $var2] ...)
array_values($array)
```

How to delete values from an array

```
$letters = array('a', 'b', 'c', 'd'); // a, b, c, d
unset($letters[2]);                   // a, b, NULL, d
unset($letters);                      // $letters is NULL
```

How to remove NULL elements and reindex

```
$letters = array('a', 'b', 'c', 'd'); // a, b, c, d
unset($letters[2]);                   // a, b, NULL, d
$letters = array_values($letters);    // a, b, d
```

How to use array elements with variable substitution

```
$name = array ('Ray', 'Harris');
echo "First Name: $name[0]";     // First Name: Ray
echo "First Name: {$name[0]}";   // First Name: Ray
```


Functions for loops that work with arrays

```
count($array)
end($array)
key($array)
isset($var)
```

Code that stores 10 random numbers in an array

```
$numbers = array();
for ($i = 0; $i < 10; $i++) {
    $numbers[] = mt_rand(1, 100);
}
```

Code that displays the elements of an array

```
$numbers_string = '';
for ($i = 0; $i < count($numbers); $i++) {
    $numbers_string .= $numbers[$i] . ' ';
}
echo $numbers_string;
```


Computing the sum and average of an array

```
$prices = array(141.95, 212.95, 411, 10.95);
$sum = 0;
for ($i = 0; $i < count($prices); $i++) {
    $sum += $prices[$i];
}
$average = $sum / count($prices);
```

How to skip gaps in an array

```
$numbers = array(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);
unset($numbers[2], $numbers[6]);
end($numbers);
$last = key($numbers);
$numbers_string = '';
for($i = 0; $i <= $last; $i++) {
    if (isset($numbers[$i])) {
        $numbers_string .= $numbers[$i] . ' ';
    }
}
echo $numbers_string;       // Displays: 1 2 4 5 6 8 9 10
```


The syntax for creating an associative array

```
array([key1 => value1, key2 => value2, ... ])
```

How to create an associative array of tax rates

With one statement

```
$tax_rates =
    array('NC' => 7.75, 'CA' => 8.25, 'NY' => 8.875);
```

With multiple statements

```
$tax_rates = array();
$tax_rates['NC'] = 7.75;
$tax_rates['CA'] = 8.25;
$tax_rates['NY'] = 8.875;
```


How to create an associative array of codes

With one statement

```
$country_codes =
    array('DEU' => 'Germany',
          'JPN' => 'Japan',
          'ARG' => 'Argentina',
          'USA' => 'United States');
```

With multiple statements

```
$country_codes = array();
$country_codes['DEU'] = 'Germany';
$country_codes['JPN'] = 'Japan';
$country_codes['ARG'] = 'Argentina';
$country_codes['USA'] = 'United States';
```


How to create an associative array of extensions

```
$ext = array();
$ext[10] = 'Sales';
$ext[13] = 'Customer Service';
$ext[16] = 'Returns';
$ext[18] = 'Warehouse';
```

An array that contains integer and string indexes

```
$employees = array();
$employees[0] = 'Mike';
$employees[1] = 'Anne';
$employees[2] = 'Judy';
$employees['senior'] = 'Mike';
$employees['newest'] = 'Pren';
```

How to use the print_r() function to view an array

```
print_r($tax_rates);
// Array ( [NC] => 7.75 [CA] => 8.25 [NY] => 8.875 )
```


How to set a value with a specific key

```
$name = array('first' => 'Ray', 'last' => 'Harris');
$name['middle'] = 'Thomas';
```

What happens when you omit the key when adding a value

```
$name = array('first' => 'Ray', 'last' => 'Harris');
$name[] = 'Thomas';                // key is 0
```

How to get a value at a specified key

```
$name = array('first' => 'Ray', 'last' => 'Harris');
$first_name = $name['first'];
$last_name = $name['last'];
```


How to delete values from an array

```
$name = array('first' => 'Ray', 'last' => 'Harris');
unset($name['first']);   // delete an element's value
unset($name);            // delete all elements
```

How to use variable substitution with elements

```
$name = array('first' => 'Ray', 'last' => 'Harris');
echo "First Name: $name['first']";   // A parse error
echo "First Name: $name[first]";     // First Name: Ray
echo "First Name: {$name['first']}"; // First Name: Ray
```

The syntax of a foreach loop

```
foreach ($array_name as [ $key => ] $value) {
    // Statements that use $key and $value
}
```

A loop that displays the values in an array

```
$tax_rates = array('NC' => 7.75, 'CA' => 8.25,
                   'NY' => 8.875);
echo '<ul>';
foreach ($tax_rates as $rate) {
    echo "<li>$rate</li>";
}
echo '</ul>';
```

The result displayed in a browser

- 7.75
- 8.25
- 8.875

A foreach loop that displays the keys and values

```
$tax_rates = array('NC' => 7.75, 'CA' => 8.25,
                   'NY' => 8.875);
echo '<ul>';
foreach ($tax_rates as $state => $rate) {
    echo "<li>$state ($rate)</li>";
}
echo '</ul>';
```

The result displayed in a browser

- NC (7.75)
- CA (8.25)
- NY (8.875)

A foreach loop that displays the values in a regular array

```
$numbers = array(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);
unset($numbers[2], $numbers[6]);
$numbers_string = '';
foreach($numbers as $number) {
    $numbers_string .= $number . ' ';
}
echo $numbers_string;      // Displays: 1 2 4 5 6 8 9 10
```


Key terms

- Foreach statement
- Foreach loop

Functions for creating arrays

```
range($lo, $hi [, $step])
array_fill($start, $count, $value)
array_pad($array, $size, $value)
array_merge($array1, $array2, ...)
array_slice($array, $index
    [, $len [, $keys]])
array_splice($array, $index
    [, $len [, $new]])
```


How to create an array with a range of values

```
$numbers = range(1, 4);           // 1, 2, 3, 4
$numbers = range(10, 22, 4);      // 10, 14, 18, 22
```

How to fill and pad an array

```
$numbers = array_fill(0, 5, 1);     // 1, 1, 1, 1, 1
$numbers = array_pad($numbers, 10, 0);
// 1, 1, 1, 1, 1, 0, 0, 0, 0, 0
```

How to merge two arrays

```
$employees = array('Mike', 'Anne');
$new_hires = array('Ray', 'Pren');
$employees = array_merge($employees, $new_hires);
print_r($employees);
// Mike, Anne, Ray, Pren
```

How to slice one array from another

```
$employees = array('Mike', 'Anne', 'Ray', 'Pren');
$new_hires = array_slice($employees, 2);
Print_r($new_hires);     // Ray, Pren
```

How to splice two arrays together

```
$employees = array('Mike', 'Anne', 'Joel');
$new_hires = array('Ray', 'Pren');
array_splice($employees, 1, 2, $new_hires);
print_r($employees);     // Mike, Ray, Pren
```


Functions for working with queues and stacks

```
array_push($array, $value)
array_pop($array)
array_unshift($array, $value)
array_shift($array)
```

How to work with a stack

```
$names = array('Mike', 'Joel', 'Anne');
array_push($names, 'Ray');    // Mike, Joel, Anne, Ray
$next = array_pop($names);    // Mike, Joel, Anne
echo $next;                   // displays Ray
```

How to work with a queue

```
$names = array('Mike', 'Joel', 'Anne');
array_push($names, 'Ray');    // Mike, Anne, Joel, Ray
$next = array_shift($names);  // Anne, Joel, Ray
echo $next;                   // displays Mike
```


Key terms

- stack
- last-in, first-out (LIFO)
- queue
- first-in, first-out (FIFO)

Functions for performing math calculations

```
array_sum($array)
array_product($array)
```

How to add all values in an array

```
$prices = array(141.95, 212.95, 411, 10.95);
$sum = array_sum($prices);            // 776.85
```


Functions for searching arrays

```
in_array($value, $array [, $strict])
array_key_exists($key, $array)
array_search($value, $array [, $strict])
array_count_values($array)
```


How to search an array

```
$tax_rates = array('NC' => 7.75,
                   'CA' => 8.25, 'NY' => 8.875);
$is_found = in_array(7.75, $tax_rates);       // TRUE
$is_found = in_array('7.75', $tax_rates);     // TRUE
$is_found = in_array('7.75', $tax_rates, true); //FALSE
$key_exists = array_key_exists('CA', $tax_rates); //TRUE
$key = array_search(7.75, $tax_rates);        // 'NC'
```

How to count the number of occurrences of a value in an array

```
$names = array('Mike', 'Mike', 'Mike',
               'Anne', 'Joel', 'Joel');
$occurences = array_count_values($names);
echo $occurences['Mike'];            // 3
echo $occurences['Anne'];            // 1
echo $occurences['Joel'];            // 2
```


Functions for sorting arrays

```
sort($array[, $compare])
rsort($array[, $compare])
asort($array[, $compare])
arsort($array[, $compare])
ksort($array[, $compare])
krsort($array[, $compare])
```


How to sort strings in ascending order

```
$names = array('Mike', 'Anne', 'Joel', 'Ray', 'Pren');
sort($names);           // Anne, Joel, Mike, Pren, Ray
```

How to sort numbers in ascending order

```
$numbers = array(520, '33', 9, '199');
sort($numbers, SORT_NUMERIC);       // 9, 33, 199, 520
```

How to sort in descending order

```
$names = array('Mike', 'Anne', 'Joel', 'Ray', 'Pren');
rsort($names);          // Ray, Pren, Mike, Joel, Anne
```

How to sort an associative array

```
$tax_rates = array('NC' => 7.75,
                   'NY' => 8.875, 'CA' => 8.25);
asort($tax_rates);     // sorts by value (ascending)
ksort($tax_rates);     // sorts by key (ascending)
arsort($tax_rates);    // sorts by value (descending)
krsort($tax_rates);    // sorts by key (descending)
```


Functions for modifying arrays

```
array_unique($array[, $compare])
array_reverse($array[, $keys])
shuffle($array)
array_rand($array[, $count])
```


How to modify an array

```
$names = array('Mike', 'Mike', 'Mike',
               'Anne', 'Joel', 'Joel');
$names = array_unique($names);      // Mike, Anne, Joel
$names = array_reverse($names);     // Joel, Anne, Mike
shuffle($names);        // Mike, Joel, Anne (for example)
```

How to modify an associative array

```
$tax_rates = array('NC' => 7.75,
                   'NY' => 8.875, 'CA' => 8.25);
$tax_rates = array_reverse($tax_rates, true);
```

How to get random keys from an array

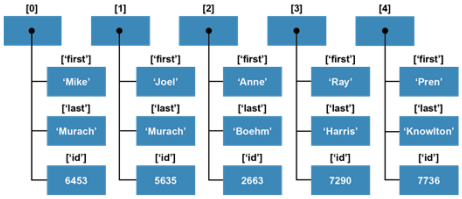
```
$names = array('Mike', 'Anne', 'Joel', 'Ray', 'Pren');
$key = array_rand($names);           // 2 (for example)
$names_rand = array_rand($names, 3);
// 0, 1, 3 (for example)
```


How to shuffle and deal a deck of cards

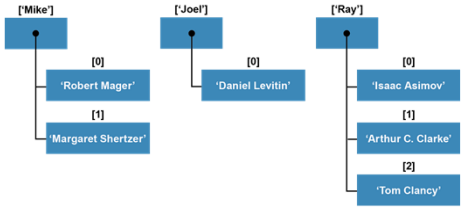
```
// Create the deck of cards
$faces = array('2', '3', '4', '5', '6', '7', '8',
               '9', 'T', 'J', 'Q', 'K', 'A');
$suits = array('h', 'd', 'c', 's');
$cards = array();
foreach($faces as $face) {
    foreach($suits as $suit) {
        $cards[] = $face . $suit;
    }
}

// Shuffle the deck and deal the cards
shuffle($cards);
$hand = array();
for ($i = 0; $i < 5; $i++) {
    $hand[] = array_pop($cards);
}
echo implode(', ', $hand);
// 9c, 6d, Ks, 4c, 7h (for example)
```

A simple array

A rectangular array of arrays



A jagged array of arrays



Key terms

- Array of arrays
- Two-dimensional array
- Rectangular array
- Jagged array

Code that creates an array of arrays

```
$times_table = array();
for ($i = 0; $i <= 12; $i++) {   // add 13 elements that
    $times_table[$i] = array();  // contain empty arrays
}
```

Code that adds values to the array of arrays

```
for ($i = 0; $i <= 12; $i++) {
    for ($j = 0; $j <= 12; $j++) {
        $times_table[$i][$j] = $i * $j;
    }
}
```

Code that refers to elements in the array of arrays

```
echo $times_table[4][3];         // displays 12
echo $times_table[7][6];         // displays 42
```


Code that creates a cart array

```
$cart = array();
```

Adding an associative array to the cart array

```
$item = array();       // create an empty item array
$item['itemCode'] = 123;
$item['itemName'] = "Visual Basic";
$item['itemCost'] = 57.5;
$item['itemQuantity'] = 5;
$cart[] = $item;       // add item array to cart array
```

Adding another associative array to the cart array

```
$item = array();       // create an empty item array
$item["itemCode"] = 456;
$item["itemName"] = "Java";
$item["itemCost"] = 59.5;
$item["itemQuantity"] = 2;
$cart[]  = $item;      // add item array to cart array
```

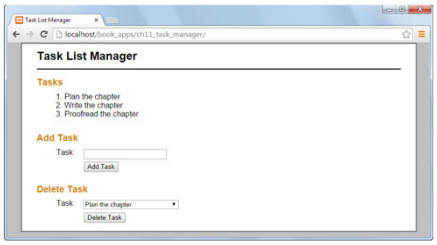
Referring to the elements in the array of arrays

```
echo $cart[0]["itemCode"];       // displays 123
echo $cart[1]["itemName"];       // displays Java
```


A more concise way to create an array of arrays

```
$cart = array(array('itemCode' => 123,
                    'itemName' => 'Visual Basic',
                    'itemCost' => 57.5,
                    'itemQuantity' => 5),
              array('itemCode' => 456,
                    'itemName' => 'Java',
                    'itemCost' => 59.5,
                    'itemQuantity' => 2));
```


The Task List Manager application



The index.php file

```
<?php
$task_list = filter_input(INPUT_POST, 'tasklist', FILTER_DEFAULT,
                          FILTER_REQUIRE_ARRAY);
if ($task_list === NULL) {
    $task_list = array();
}

$action = filter_input(INPUT_POST, 'action');
$errors = array();
```


The index.php file (continued)

```
switch( $action ) {
    case 'add':
        $new_task = filter_input(INPUT_POST, 'task');
        if (empty($new_task)) {
            $errors[] = 'The new task cannot be empty.';
        } else {
            $task_list[] = $new_task;
        }
        break;
    case 'delete':
        $task_index = filter_input(INPUT_POST, 'taskid',
        FILTER_VALIDATE_INT);
        if ($task_index === NULL || $task_index === FALSE) {
            $errors[] = 'The task cannot be deleted.';
        } else {
            unset($task_list[$task_index]);
            $task_list = array_values($task_list);
        }
        break;
}
include('task_list.php');
?>
```


The task_list.php file

```
<!DOCTYPE html>
<html>
<head>
    <title>Task List Manager</title>
    <link rel="stylesheet" type="text/css" href="main.css">
</head>
<body>
    <header>
        <h1>Task List Manager</h1>
    </header>
    <main>
```

The task_list.php file (continued)

```
<!-- part 1: the errors -->
<?php if (count($errors) > 0) : ?>
<h2>Errors</h2>
<ul>
    <?php foreach($errors as $error) : ?>
    <li><?php echo htmlspecialchars($error); ?></li>
    <?php endforeach; ?>
</ul>
<?php endif; ?>

<!-- part 2: the tasks -->
<h2>Tasks</h2>
<?php if (count($task_list) == 0) : ?>
    <p>There are no tasks in the task list.</p>
<?php else: ?>
    <ul>
    <?php foreach( $task_list as $id => $task ) : ?>
        <li><?php echo $id + 1 . '. ' .
                htmlspecialchars($task); ?></li>
    <?php endforeach; ?>
    </ul>
<?php endif; ?>
<br>
```


The task_list.php file (continued)

```
<!-- part 3: the add form -->
<h2>Add Task</h2>
<form action="." method="post" >
    <?php foreach( $task_list as $task ) : ?>
      <input type="hidden" name="tasklist[]"
             value="<?php echo htmlspecialchars($task); ?>">
    <?php endforeach; ?>
    <input type="hidden" name="action" value="add">
    <label>Task:</label>
    <input type="text" name="task"><br>
    <label> </label>
    <input type="submit" value="Add Task"><br>
</form>
<br>
```


The task_list.php file (continued)

```
<!-- part 4: the delete form -->
<?php if (count($task_list) > 0) : ?>
<h2>Delete Task</h2>
<form action="." method="post" >
    <?php foreach( $task_list as $task ) : ?>
      <input type="hidden" name="tasklist[]"
             value="<?php echo htmlspecialchars($task); ?>">
    <?php endforeach; ?>
    <input type="hidden" name="action" value="delete">
    <label>Task:</label>
    <select name="taskid">
        <?php foreach( $task_list as $id => $task ) : ?>
            <option value="<?php echo $id; ?>">
                <?php echo htmlspecialchars($task); ?>
            </option>
        <?php endforeach; ?>
    </select>
    <br>
    <label> </label>
    <input type="submit" value="Delete Task">
</form>
    <?php endif; ?>
  </main>
</body></html>
```