Escola Pia
Santa Anna-Mataró

PHP Array

INTRODUCTION PHP

Array: Definició

```
<?php

    $arr = array("Egg", "Tomato", "Beans");

  ?>
```

- An array stores multiple values in one single variable
- An array can hold many values under a single name, and you can access the values by referring to **an index number**
- In PHP, the array() function is used to create an array:

array();

Array: Definició

```
<?php

    $arr = array("Egg", "Tomato", "Beans ", 5, 125, “Potatoes" );

  ?>
```

- Use "" when adding strings, and just enter the number when adding integers.
- You must always remember, however, that each item in an array must be separated by a comma: ,
- The index can be assigned automatically (index always starts at 0) or the index can be assigned manually

$cars = array("Volvo", "BMW", "Toyota");	$cars[0] = "Volvo"; $cars[1] = "BMW"; $cars[2] = "Toyota";

Array: Accés als valors

PHP is a very flexible language. When accessing arrays by offset, you can actually use two different types of syntax:

- the [] syntax we've covered,
- or you can use curly braces {}.

```
<?php
 $myArray = array("do", "re", "mi");
 print $myArray{2};
 print $myArray[0]};
 // prints "mi“, “do”;
?>
```

Array: Accés als valors

UPDATE VALUES

```
<?php
$myArray = array("red", "blue", "yellow");
echo $myArray[1];
// outputs "blue"
$myArray[1] = "green";
echo $myArray[1];
// outputs "green"
?>
```

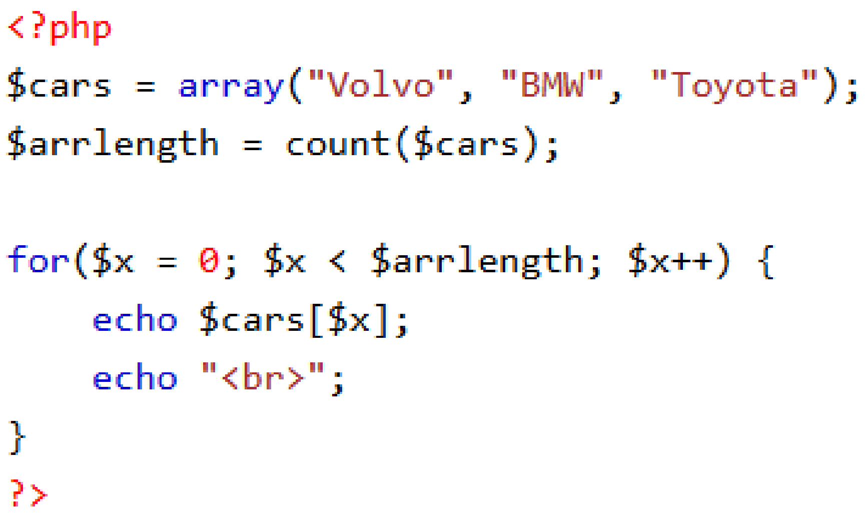
DELETE VALUES & COMPLETELY

```
<?php
 $array = array("red", "blue", "green");
 unset($array[2]);
?>
```

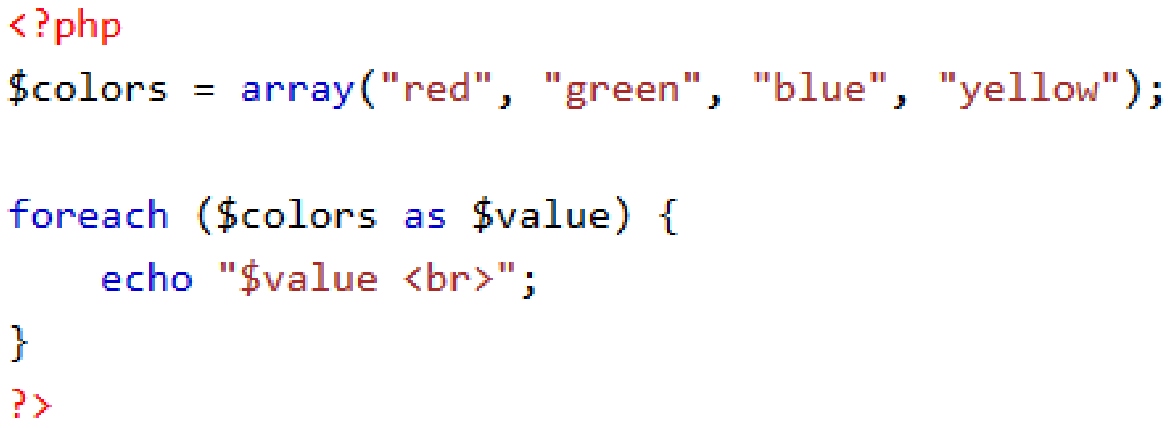
You can even delete the whole array:

```
<?php
 unset($array);
?>
```

Array: Bucles



```php
<?php
$cars = array("Volvo", "BMW", "Toyota");
$arrlength = count($cars);

for($x = 0; $x < $arrlength; $x++) {
    echo $cars[$x];
    echo "<br>";
}
?>
```



```php
<?php
$colors = array("red", "green", "blue", "yellow");

foreach ($colors as $value) {
    echo "$value <br>";
}
?>
```

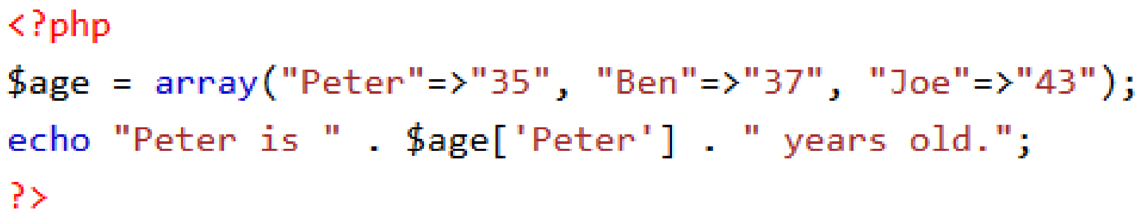
Associative Arrays

Associative arrays are arrays that use named keys that you assign to them.

There are two ways to create an associative array:

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");	$age['Peter'] = "35"; $age['Ben'] = "37"; $age['Joe'] = "43";

Associative Arrays

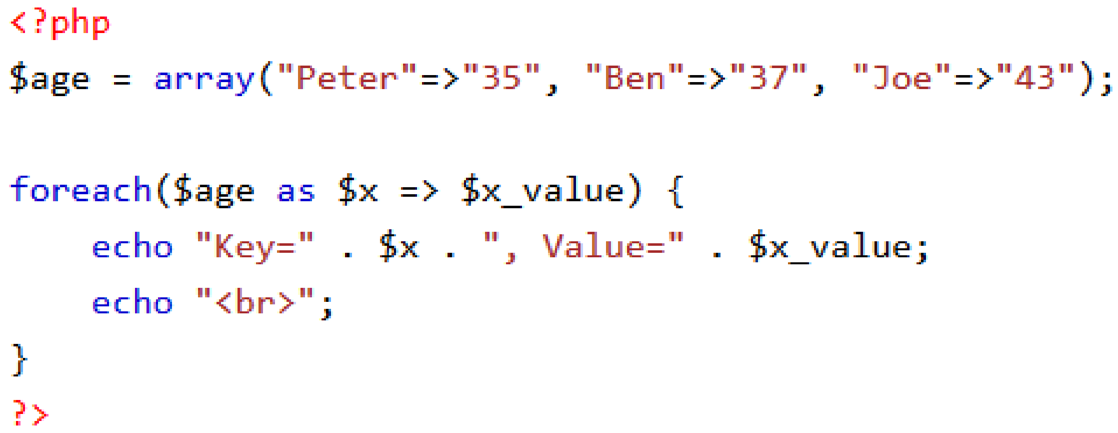


```
<?php
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
echo "Peter is " . $age['Peter'] . " years old.";
?>
```

Podem accedir als valors per la clau (key) o per l’índex

echo $age[‘Ben’]; //prints ‘37’

echo $age[1]; // prints ‘37’

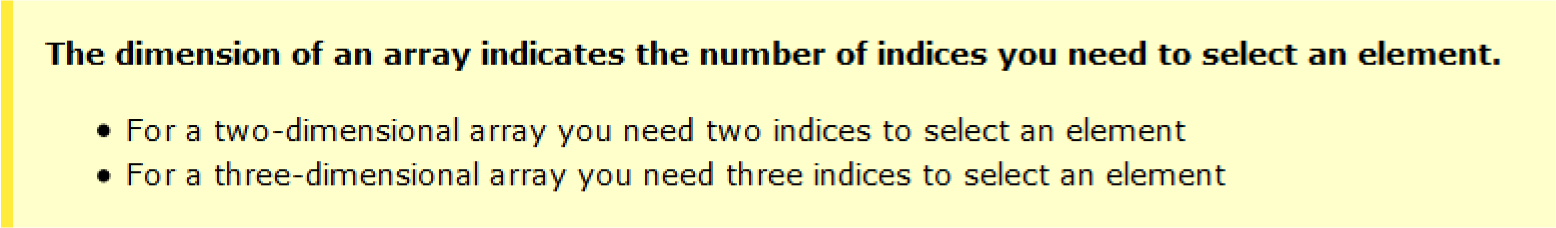


```
<?php
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

foreach($age as $x => $x_value) {
    echo "Key=" . $x . ", Value=" . $x_value;
    echo "<br>";
}
?>
```

Multidimensional Arrays

A multidimensional array is an array containing one or more arrays.

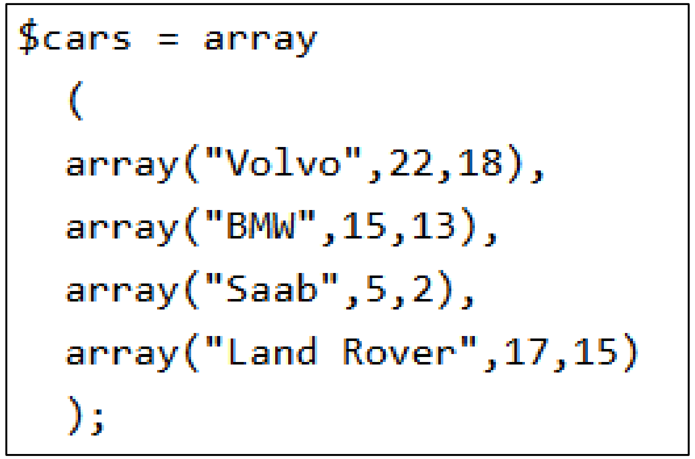
PHP understands multidimensional arrays that are two, three, four, five, or more levels deep. However, arrays more than three levels deep are hard to manage for most people.



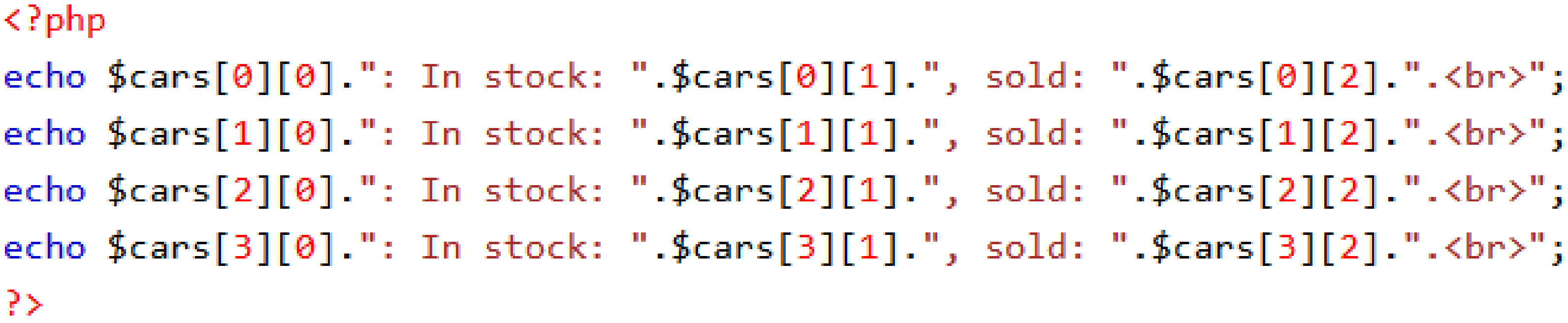
The dimension of an array indicates the number of indices you need to select an element.

- For a two-dimensional array you need two indices to select an element
- For a three-dimensional array you need three indices to select an element

Multidimensional Arrays

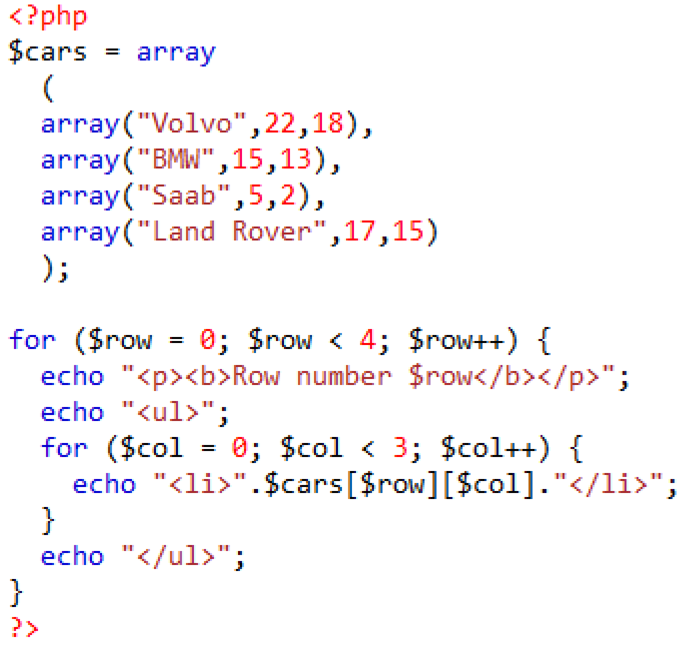


```
$cars = array
  (
  array("Volvo",22,18),
  array("BMW",15,13),
  array("Saab",5,2),
  array("Land Rover",17,15)
  );
```



```
<?php
echo $cars[0][0].": In stock: ".$cars[0][1].", sold: ".$cars[0][2].".<br>";
echo $cars[1][0].": In stock: ".$cars[1][1].", sold: ".$cars[1][2].".<br>";
echo $cars[2][0].": In stock: ".$cars[2][1].", sold: ".$cars[2][2].".<br>";
echo $cars[3][0].": In stock: ".$cars[3][1].", sold: ".$cars[3][2].".<br>";
?>
```

Multidimensional Arrays

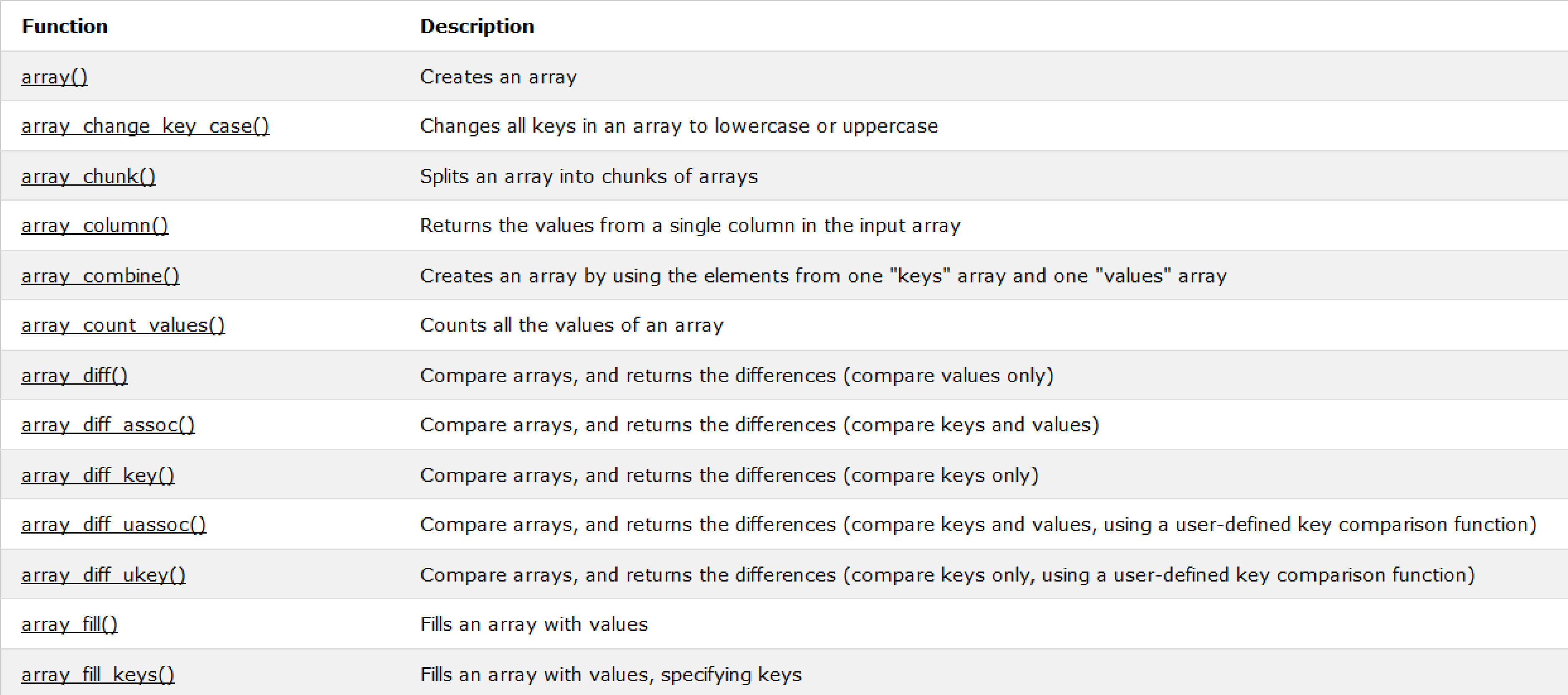


```
<?php
$cars = array
  (
  array("Volvo",22,18),
  array("BMW",15,13),
  array("Saab",5,2),
  array("Land Rover",17,15)
  );

for ($row = 0; $row < 4; $row++) {
  echo "<p><b>Row number $row</b></p>";
  echo "<ul>";
  for ($col = 0; $col < 3; $col++) {
    echo "<li>".$cars[$row][$col]."</li>";
  }
  echo "</ul>";
}
?>
```

```
<?php

    $cars = array  (array("Volvo",22,18),
                          array("BMW",15,13),
                           array("Saab",5,2),
                           array("Land Rover",17,15)
                          );

  foreach ($cars as $car) {

   foreach ($car as $value) {
          echo "<li>$value</li>"; }

    }



   ?>
```

https://www.w3schools.com/php/php_ref_array.asp
http://php.net/manual/en/ref.array.php

Array Functions



Function	Description
array()	Creates an array
array_change_key_case()	Changes all keys in an array to lowercase or uppercase
array_chunk()	Splits an array into chunks of arrays
array_column()	Returns the values from a single column in the input array
array_combine()	Creates an array by using the elements from one "keys" array and one "values" array
array_count_values()	Counts all the values of an array
array_diff()	Compare arrays, and returns the differences (compare values only)
array_diff_assoc()	Compare arrays, and returns the differences (compare keys and values)
array_diff_key()	Compare arrays, and returns the differences (compare keys only)
array_diff_uassoc()	Compare arrays, and returns the differences (compare keys and values, using a user-defined key comparison function)
array_diff_ukey()	Compare arrays, and returns the differences (compare keys only, using a user-defined key comparison function)
array_fill()	Fills an array with values
array_fill_keys()	Fills an array with values, specifying keys

Escola Pia
Santa Anna-Mataró