

PHP - Arrays simples

DAW-MP07-UF1 - Exercici de Desenvolupament web en entorn servidor.

Un array a PHP és realment un mapa ordenat. Per aquest motiu podem tractar l'array indexant el numèricament tal i com fem en altres llenguatges. En aquest exercici crearem un array, afegirem valors, en treurem i el pintarem:

```
<?php
// creo un array amb 3 elements
$a = array( 5,7,11);
print_r( $a );
echo "<br>";

//afegeixo més elements a l'array
$a[] = 13;
$a[] = 17;
print_r( $a );
echo "<br>";

//encara un altre
array_push ( $a, 23);
print_r( $a );
echo "<br>";

//pinto elements de l'array
echo "El valor de del tercer element de l'array és " . $a[2];
echo "<br>";

unset($a[0]);
unset($a[1]); // el valor 7 seguia a la posició 1
print_r( $a );
echo "<br>";
?>
```

1. Quina és la instrucció a PHP per a crear un array?
2. Es pot crear un array buit a PHP ó sempre s'hi han de posar valors a la creació?
3. Si li prenem el tipus a l'array amb la funció *gettype()*, quin tipus obtenim?
4. Com puc fer per afegir un element a l'array a partir de la darrera posició coneguda? Esmenta dues maneres diferents.
5. Com s'accedeix a un element d'un array?
6. Com s'esborra un element d'un array?

7. Fixat que el valor 7, emmagatzemat a la posició \$a[1], segueix exactament a la mateixa posició després d'esborrar \$a[0]. Això no seria així en altres llenguatges de programació, però a PHP els arrays, realment, són mapes ordenats. Farem el següent exercici per comprendre aquest concepte.

1. \$a = array (n,m,p);
2. Si que es pot crear un array buit.

```
Array ( )
```

```
Array ( [0] => 13 [1] => 17 )
```

```
Array ( [0] => 13 [1] => 17 [2] => 23 )
```

El valor de del tercer element de l'array és 23

```
Array ( [2] => 23 )
```

- 3.

La variable \$a conté el valor Array i és del tipus array

4. `$a[] = 13;` posant directament el valor, així automàticament te'l posa a la última posició coneguda.

`array_push ($a, 23);` la funció array_push() afegeix elements a la última posició.

- 5.

`$a[2];` posant la variable de l'array i la posició determinada entre claudàtors.

6. A partir de la funció unset(nom de la variable i la posició desitjada).

```
unset($a[0]);
```