



3. Arrays en PHP

Profesor: Alejandro Amat Reina



Arrays

- Un array es un tipo de datos que nos permite almacenar varios valores
- Para acceder a un valor utilizaremos una clave.
- Las claves pueden ser números o textos (arrays asociativos).
- Si no indicamos ninguna clave, a cada elemento se le asociará una clave numérica correlativa.



Crear array

- Podemos crear un array de dos formas:
 - Usando la función array:
`$nombres = array();`
 - Usando un atajo disponible desde la versión 5.4 de PHP:
`$nombres = [];`
 - En ambos ejemplos estamos creando un array vacío.



Crear array con valores

- Si queremos asignar valores al array en el momento de crearlo:

```
$nombres = array('Sandra', 'Pedro', 'Andrea');  
$nombres = ['Sandra', 'Pedro', 'Andrea'];
```



Mostrar el contenido de un array

- Si intentamos mostrar el contenido de un array con `echo` obtendremos el texto `Array`.
- `Echo` no muestra los datos que hay dentro del array.
- En lugar de esto podemos utilizar la función `print_r`:
 - `print_r($nombres);`
- Si necesitamos darle formato a los contenidos del array, tendremos que recorrerlo con un bucle e ir mostrando elemento a elemento.



Añadiendo elementos al array

- Podemos añadir elementos al array utilizando corchetes vacíos:

```
$nombres[] = 'Raul';  
$nombres[] = 'Marta';
```

- La clave de estos elementos será el siguiente índice numérico disponible.

Acceder a los elementos del array



- Accederemos a los elementos del array indicando la clave del elemento entre corchetes:
 - **echo** \$nombres[2];
- Podemos modificar su valor:
 - \$nombres[0] = 'Sara';



Arrays asociativos

- Si el array contiene datos diversos y/o nos interesa acceder a ellos con claves más específicas que un simple índice numérico podemos utilizar arrays asociativos:

```
$alumno = array(  
    'nombre' => 'Sara',  
    'apellido' => 'García',  
    'edad' => 22  
);  
$alumno = [  
    'nombre' => 'Sara',  
    'apellido' => 'García',  
    'edad' => 22  
];  
$alumno['nombre'] = 'Verónica';  
echo $alumno['nombre'];
```




Arrays multidimensionales

- Imagina que queremos tener un array cuyos elementos son nombres de deportes.
- Queremos dividirlos en deportes de invierno y deportes de verano.
- Podemos crear un array `$deportes` que contenga dos elementos que a su vez también serán arrays.

```
$deportes = array(  
    'invierno' => array('esquí de fondo', 'hockey sobre hielo'),  
    'verano' => array('natación', 'voley playa')  
);
```

- Para acceder al primer deporte de invierno:
 - `echo $deportes['invierno'][0];`

Arrays Multidimensionales asociativos

- La segunda dimensión del array también puede ser un array asociativo:

```
$alumnos = array(  
    array(  
        'nombre' => 'Sara',  
        'edad' => 25  
    ),  
    array(  
        'nombre' => 'Pedro',  
        'edad' => 23  
    ),  
);
```

- Mostraremos el nombre del primer alumno:

- `echo $alumnos[0]['nombre'];`



Funciones de arrays

- Como siempre no tiene sentido ir viéndolas de una en una.
- Podemos consultarlas en <http://es.php.net/manual/es/ref.array.php>



Ejercicios de arrays

- *Crea una página llamada arrays.php, copia el contenido de la página index.php y resuelve los ejercicios siguientes utilizando funciones de arrays:*
 - Crea un array con los nombres de varios alumnos de la clase incluyendo el tuyo.
 - Muestra el número de elementos que tiene el array.
 - Crea una cadena que contenga los nombres de los alumnos existentes en el array separados por un espacio y muéstrala.
 - Muestra el array en un orden aleatorio distinto al que lo creaste.
 - Muestra el array ordenado alfabéticamente.
 - Muestra los alumnos cuyo nombre contenga al menos una 'a'.
 - Muestra el array en el orden inverso al que se creó.
 - Muestra la posición que tiene tu nombre en el array.

Recorrer arrays

```
<?php
$meteors = array(
    'Hoba',
    'Cape York',
    'Campo del Cielo',
    'Canyon Diablo',
);

foreach($meteors as $meteor) {
    echo $meteor;
}
```

On each pass through our foreach loop, the data in \$meteor will update with the next item in the collection.

Output

Hoba
Cape York
Campo del Cielo
Canyon Diablo

The value, our meteorite names

Recorrer arrays asociativos

```
<?php
$meteors = array(
    'Hoba' => 600000000,
    'Cape York' => 58200000,
    'Campo del Cielo' => 50000000,
    'Canyon Diablo' => 30000000,
);

foreach($meteors as $meteor) {
    echo $meteor;
}
```

Output



600000000
58200000
50000000
30000000

The value is our meteorite weight!



Recorrer claves y valores

```
<?php
$meteors = array(
    'Hoba' => 6000000000,
    'Cape York' => 58200000,
    'Campo del Cielo' => 50000000,
    'Canyon Diablo' => 30000000,
);

foreach($meteors as $name => $weight){
    echo "$name weighs $weight grams.";
}
```

Output

**Hoba weighs
6000000000 grams.**

...

**Canyon Diablo weighs
30000000 grams.**



Ejercicios de arrays

- *Crea una página llamada arrays2.php, copia el contenido de la página index.php y resuelve los ejercicios siguientes utilizando funciones de arrays:*
 - *Crea un array de alumnos donde cada elemento sea otro array que contenga el id, nombre y edad del alumno.*
 - *Crea una tabla en la que se muestren todos los datos de los alumnos.*
 - *Utiliza la función array_column para obtener un array indexado que contenga únicamente los nombres de los alumnos y muéstralo por pantalla.*
 - *Crea un array con 10 números :*
 - *utiliza la función array_sum para obtener la suma de los 10 números.*
 - *utiliza la función array_reduce para obtener el factorial de los 10 números.*