

Arrays bidimensionales

Los *arrays bidimensionales* pueden entenderse como algo muy similar a una *tabla de doble entrada*.

Cada uno de los elementos se identifica –sigue siendo válido el nombre único que se usaba en los unidimensionales – por un nombre (**\$nombre**) seguido de dos (**[]**) que contienen los *índices* (en este caso son dos índices) del array.

Los *índices* pueden ser de tipo **escalar** –equivalen al número de fila y columna que la **celda** ocupa en la tabla– o puede ser **asociativos** lo que equivaldría en alguna medida a usar como índices los *nombres de la fila y de la columna*.

¡Cuidado!

No dejes de tener en cuenta lo que hemos advertido al hablar de arrays unidimensionales.

En este supuesto, también, se empiezan a numerar los arrays escalares a partir de **CERO**.

Arrays escalares

Los elementos de un *array bidimensional* escalar pueden escribirse usando una de estas sintaxis:

```
$a[] = valor
o
$a[xx] = valor
o
$a[][xx] = valor
o también
$a[xx][yy] = valor
```

En el primero de los casos PHP asigna automáticamente como **primer índice** el valor que sigue al último asignado y, si es el **primero** que se define, le pondrá como índice **0** (CERO).

Sea cual fuere el valor de primer índice al **segundo** se le asignará **cero** ya que es en este mismo momento cuando se habrá creado el **primero** y, por tanto, *aún carecerá de elementos*.

En el segundo de los casos, asignamos un valor al **primer índice (xx)** y será el segundo quien se incremente en una **unidad** respecto al de valor más alto de todos aquellos cuyo **primer índice** coincide con el especificado.

La tercera opción es bastante similar a la anterior. Ahora se modificaría automáticamente el primer índice y se escribiría el contenido (xx) como valor del segundo.

En la cuarta de las opciones se asignan libremente cada uno de los índices (**xx** e **yy**) poniéndoles valores numéricos.

Arrays asociativos

Los elementos de un *array asociativo* se pueden escribir usando la siguiente sintaxis:

```
$a["índice1"]["índice2"] = valor
```

En este caso, los índices serán **cadenas** y se escribirán entre

Arrays bidimensionales

Como ejemplo de array bidimensional emplearemos una tabla de resultados de una *liga de fútbol* en la que intervienen **cinco equipos** que –como en toda liga que se precie– se juega a *doble partido*.

En este primer supuesto utilizaremos *arrays escalares*, por lo tanto los equipos serán identificados con números desde **cero** hasta **cuatro**.

```
<?
# rellenamos el array desde [0][0] hasta [0][4]
# la insercion automatica haria que este primero fuera [0][0]

$a[][] = " ";
# ahora pondremos cero como indice del primer array y dejemos que PHP
# nos vaya insertando automaticamente el segundo

$a[0][] = "3-2"; $a[0][] = "5-3"; $a[0][] = "7-1"; $a[0][] = "0-2";

#ahora desde [1][0] hasta [1][4]
#este primero lo dejamos como automatico en ambos indices
# de esta forma el primero tomará valor uno (siguiente al anterior)
# de forma automática
$a[][] = "0-11";
# repetimos el proceso anterior
$a[1][] = " "; $a[1][] = "2-1"; $a[1][] = "1-0"; $a[1][] = "1-2";
# y repetimos de nuevo, ahora crearia 2 como primer indice
$a[][] = "0-0";
#insertariamos los restantes valores de indice 2
$a[2][] = "1-3"; $a[2][] = " "; $a[2][] = "1-4"; $a[2][] = "2-0";
# nuevo incremento del primer indice
$a[][] = "1-0";
# rellenamos
$a[3][] = "6-3"; $a[3][] = "14-3 "; $a[3][] = " "; $a[3][] = "1-0";
# nuevo y ultimo incremento de primer indice
$a[][] = "1-1";
# rellenamos de nuevo
$a[4][] = "2-3"; $a[4][] = "0-1 "; $a[4][] = "1-1"; $a[4][] = " ";

# como verás el proceso no tiene complicaciones, pero ... pesadillo si es
# ¿verdad que si tuviéramos una base de datos sería más fácil?
# estamos en ello, todo se andará...

# tendríamos que ver esos valores pero.. escribir "a mano"
# una tabla puede ser una tortura, así que mejor introducimos
# una bucle, otro recurso que estudiaremos pronto
# para esa labor repetitiva de mostrar en una tabla
# todos los datos del array

# Sería algo como esto
# creamos la etiqueta de apertura de una tabla

print("<TABLE BORDER=2>");
# ahora dos bucles anidados (rojo uno, magenta el otro)
# para rellenar las celdas de cada fila (el magenta)
# y para insertar las etiquetas <TR> utilizaremos el rojo

for ($i=0;$i<5;$i++){
    print("<tr>");
    for ($j=0;$j<5;$j++) {
        print("<td>".$a[$i][$j]."</td>");
    }
}

#ponemos la etiqueta de cierre de la tabla

print("</table>");
?>
```

ejemplo20.php

Utilizando el script anterior, con ligeros *retoques estéticos*, hemos construido esta tabla:

Todos los resultados de la <i>liguilla</i>					
Índice	0	1	2	3	4
0		3-2	5-3	7-1	0-2
1	0-11		2-1	1-0	1-2
2	0-0	1-3		1-4	2-0
3	1-0	6-3	14-3		1-0
4	1-1	2-3	0-1	1-1	

comillas.

Arrays mixtos

PHP permite utilizar también arrays *mixtos*. Sería este el caso de que uno de ellos fuera escalar y el otro asociativo.

Igual que ocurría con los unidimensionales, también aquí podemos utilizar valores de variables como índices.

Con el mismo procedimiento –en este caso hemos usado resultados diferentes– hemos construido esta otra tabla

Resultados de la liguilla					
Indice	<i>Juvençia</i>	<i>Mosconia</i>	<i>Canicas</i>	<i>Condal</i>	<i>Piloñesa</i>
<i>Juvençia</i>		3-2	5-3	7-1	0-2
<i>Mosconia</i>	0-11		2-1	1-0	1-2
<i>Canicas</i>	0-0	1-3		1-4	2-0
<i>Condal</i>	1-0	6-3	14-3		1-0
<i>Piloñesa</i>	1-1	2-3	0-1	1-1	

¿Por qué no intentas modificar el script y tratas de reproducir estas tablas? Desde luego, es *sólo una sugerencia*.

Anterior Indice Siguiente

