

# DESARROLLO WEB SERVIDOR (DWS)



CURSO 2017-2018  
10 / 12 / 2017

---

PRÁCTICA TEMA 2

## LENGUAJES DE MARCAS

CON CODIGO EMBEBIDO

Por:  
José Luis Ferrete Olarte

**Licencia Creative Commons BY-NC-SA**

Los contenidos de este documento se publican bajo licencia CC BY-NC-SA: Creative Commons Reconocimiento – NoComercial – CompartirIgual.

**Cedemos los siguientes derechos sobre la obra:**

Derecho de reproducción, distribución y comunicación pública sobre la obra.

Derecho a incorporarla en una o más obras conjuntas o bases de datos y para su reproducción en tanto que incorporada a dichas obras conjuntas o bases de datos.

Derecho para efectuar cualquier transformación sobre la obra y crear y reproducir obras derivadas.

Derecho de distribución y comunicación pública de copias o grabaciones de la obra, como incorporada a obras conjuntas o bases de datos.

Derecho de distribución y comunicación pública de copias o grabaciones de la obra, por medio de una obra derivada.

**Siempre que lo hagas con estas condiciones:**

Reconocer la autoría, especificando la firma y el autor que lo publica (*"D. José Luis Ferrete Olarte"*). Si es en formato digital, debes añadir un enlace al contenido original.

Compartir bajo la misma licencia. Si reproduces o remezclas esta obra, sólo puedes distribuir la obra generada bajo una licencia como ésta (con las mismas condiciones). Hemos realizado este trabajo con fines educativos y queremos que, si nos utilizas como base para tus creaciones, también contribuyas a la comunidad difundiéndolas con licencias libres. Así, aprenderemos todos juntos.

No se permite un uso comercial de la obra original, ni de las posibles obras derivadas.

Ante cualquier duda sobre las condiciones de cesión de derechos, consultar: <https://creativecommons.org/licenses/by-nc-sa/2.5/es/legalcode.es>

## INDICE

I.	Introducción .....	3
II.	Ejercicios.....	4
A.	Crea una página Web que genere un número aleatorio entre 1 y 100 e imprima los números impares comprendidos entre dicho número y cero. ....	4
B.	Crea una función que reciba dos números y devuelva el número de números pares comprendidos entre ellos. Además, se deben imprimir dichos números. Llama a la función para comprobar su funcionamiento. ....	5
C.	Crea una página PHP con un procedimiento llamado “cuadrado”, que recibe un argumento (un número entero que representa la longitud del lado de un cuadrado). La función debe imprimir por pantalla el valor de la superficie y del perímetro del cuadrado. Declara un variable con el valor deseado y haz una llamada al procedimiento para comprobar su funcionamiento. ....	6
D.	Crea una función que recibe un parámetro y que calcule la suma de los números inferiores a él. Llámala desde el cuerpo del script e imprime el resultado por pantalla.....	7
E.	Crea una página php que defina una función “cuenta” que recibe dos parámetros. El primero de ellos será un array de números enteros y el segundo parámetro una variable numérica. La función se encargará de contar el número de veces que se repite el número que recibe (segundo parámetro) dentro del array. A continuación, en el cuerpo de la página, crea un array y rellénalo con números aleatorios entre 1 y 10 y llama a la función para comprobar el resultado. ....	8
F.	Busca una función que permita eliminar los valores repetidos de un array. Modifica el ejercicio anterior de forma que, tras la llamada a la función “cuenta”, llame a la función que elimina los valores repetidos, y vuelva a llamar a la función “cuenta”, para ver el resultado. ....	9
G.	Crea un script que defina una matriz de dos dimensiones, que recoja los datos de "Nombre", "Apellidos", "Idioma" y “Nota Media” para los alumnos de una clase con un total de 5 alumnos. En dicho script define un procedimiento “comprueba”, que reciba un número entero (identificador del alumno en la matriz anterior) y recorra la matriz que almacena los datos de dicho alumno, mostrándolos por pantalla. Llámalo desde el cuerpo de la página. Crea otra función “estadísticas”, que reciba una matriz y devuelva la media de la clase y el número de aprobados. Llámala desde el cuerpo de la página. ....	10
III.	Bibliografía .....	12

## I. Introducción

El objetivo fundamental de esta práctica es profundizar en el aprendizaje del lenguaje de programación PHP, mediante la resolución de una serie de ejercicios. Trataremos de afrontarlos comprendiendo la metodología propuesta para la resolución de los mismos y tratando de interiorizar los nuevos conceptos, así como el lenguaje propio a utilizar.

PHP está enfocado principalmente a la programación de scripts del lado del servidor, por lo que se puede hacer cualquier cosa que pueda hacer otro programa CGI, como recopilar datos de formularios, generar páginas con contenidos dinámicos, o enviar y recibir cookies. Aunque PHP puede hacer mucho más.

## II. Ejercicios

A. Crea una página Web que genere un número aleatorio entre 1 y 100 e imprima los números impares comprendidos entre dicho número y cero.

```
<?php
// Generamos el número aleatorio entre 1 y 100
$numAle= rand(1,100);
/*Recorremos los numeros comprendidos entre el numero aleatorio y el 0,
comprobando si se trata de un número impar
*/
echo("Los números impares comprendidos entre $numAle y 0 son: <br/>");
for($i=$numAle;$i>=0;$i--){
    if($i%2!=0){
        echo("$i<br/>");
    }else{
    }
}
?>
```

### Ejercicio 1

Los números impares comprendidos entre 6 y 0 son:

5  
3  
1

B. Crea una función que reciba dos números y devuelva el número de números pares comprendidos entre ellos. Además, se deben imprimir dichos números. Llama a la función para comprobar su funcionamiento.

```
<?php

function numPar($num1,$num2){

    $contador=0;
    $numMax = max($num1,$num2);
    $numMin = min($num1,$num2);
    $arrNum = array();

    for($numMax;$numMax>=$numMin;$numMax--){
        if($numMax%2==0){
            array_push($arrNum,$numMax);
            $contador++;
        }
    }
    $msg="La cantidad de números pares comprendidos entre los numeros elegidos
es de $contador <br/>" . "Son los siguientes: " . implode(",",$arrNum);
    return $msg;

}
$num1=rand(1,100);
echo("El primer número es: $num1<br/>");
$num2=rand(1,100);
echo("El segundo número es: $num2<br/>");
echo(numPar($num1,$num2));
?>
```

### Ejercicio 2

El primer número es: 77  
El segundo número es: 64  
La cantidad de números pares comprendidos entre los numeros elegidos es de 7  
Son los siguientes: 76,74,72,70,68,66,64

C. Crea una página PHP con un procedimiento llamado “cuadrado”, que recibe un argumento (un número entero que representa la longitud del lado de un cuadrado). La función debe imprimir por pantalla el valor de la superficie y del perímetro del cuadrado. Declara un variable con el valor deseado y haz una llamada al procedimiento para comprobar su funcionamiento.

```
<?php
```

```
function cuadrado($lado){  
    $area=$lado*$lado;  
    $perimetro=$lado*4;  
    echo("El área del cuadrado de lado $lado es $area y su perimetro es de  
$perimetro <br/>");  
}  
$lado=rand(1,100);  
echo ("El valor del lado generado aleatoriamente es de: $lado <br/>");  
cuadrado($lado);  
?>
```

### Ejercicio 3

El valor del lado generado aleatoriamente es de: 87  
El área del cuadrado de lado 87 es 7569 y su perimetro es de 348

D. Crea una función que recibe un parámetro y que calcule la suma de los números inferiores a él. Llámala desde el cuerpo del script e imprime el resultado por pantalla.

```
<?php
function inferior($numero){
    $sumatorio=0;
    for($i=1;$i<$numero;$i++){
        $sumatorio=$sumatorio+($i);
    }
    return $sumatorio;
}
$num3=rand(1,100);
echo("El número generado aleatoriamente es $num3 <br/>");
echo("La suma de los valores inferiores al número $num3 es de " . inferior($num3)
. "<br/>");

?>
```

#### Ejercicio 4

El número generado aleatoriamente es 48  
La suma de los valores inferiores al número 48 es de 1128



E. Crea una página php que defina una función “cuenta” que recibe dos parámetros. El primero de ellos será un array de números enteros y el segundo parámetro una variable numérica. La función se encargará de contar el número de veces que se repite el número que recibe (segundo parámetro) dentro del array. A continuación, en el cuerpo de la página, crea un array y rellénalo con números aleatorios entre 1 y 10 y llama a la función para comprobar el resultado.

```
<?php
function cuenta($arrValores,$numRep){
    /*Buscando en W3CSchools (
https://www.w3schools.com/php/func_array_count_values.aspx ) en la documentación
de
    PHP ( http://php.net/manual/es/function.array-count-values.php ), encontré
la función (array_count_values())
    que cuenta todos los valores dentro de una array.
    Devuelve una array asociativa, donde las claves son los valores originales
de la array, y los valores el número de coincidencias
    Emplear este método, nos ahorra muchas líneas de código*/
    $contador=array_count_values($arrValores);
    $resultado=$contador[$numRep];
    return $resultado;
}

$arrValores=array();
echo("La lista de valores de la array es: ");
for($i=0;$i<50;$i++){
    $num4=rand(1,10);
    array_push($arrValores,$num4);
    echo (" $num4 ");
}
$num5= rand(1,10);
$vecesRep = cuenta($arrValores,$num5);
echo("<br/> El numero de veces que obtenemos repetido $num5 es: $vecesRep");
?>
```

### Ejercicio 5

La lista de valores de la array es: 2 10 9 10 8 2 7 3 1 10 4 6 9 2 5 7 5 2 2 7 8 2 1 3 2 5 2 9 4 1 1 10 8 5 1 10 2 6 5 6 1 10 6 9 2 5 9 6 8 7  
El numero de veces que obtenemos repetido 5 es: 6

F. Busca una función que permita eliminar los valores repetidos de un array. Modifica el ejercicio anterior de forma que, tras la llamada a la función “cuenta”, llame a la función que elimina los valores repetidos, y vuelva a llamar a la función “cuenta”, para ver el resultado.

```
<?php
/* La función array_unique ( http://php.net/manual/es/function.array-unique.php ) elimina valores
duplicados de un array */
/*Al estar ejecutando la función “cuenta” varias veces en el mismo documento, no se puede
redefinir por eso está comentada. Se toma del ejercicio anterior, en el que está definida. */
/*function cuenta($arrValores,$numRep){
    $contador=array_count_values($arrValores);
    $resultado=$contador[$numRep];
    return $resultado;
}*/

function criba($arrValores){
    $resultadoCriba = array_unique($arrValores);
    return $resultadoCriba;
}

$arrValores=array();
echo("La lista de valores de la array es: ");
for($i=0;$i<50;$i++){
    $num4=rand(1,10);
    array_push($arrValores,$num4);
    echo (" $num4 ");
}

$num5= rand(1,10);
$vecesRep = cuenta($arrValores,$num5);
echo("<br/> El numero de veces que obtenemos repetido $num5 es: $vecesRep. <br/>");
//Modificamos el ejercicio 5 y ejecutamos la función criba para eliminar valores duplicados
$arrCriba = criba($arrValores);
//Ahora comprobamos los resultados obtenidos con la array cribada
echo("La lista de valores tras la criba de la array es: ".implode(",",$arrCriba));
$vecesRep = cuenta($arrCriba,$num5);
echo("<br/>Tras la criba, el numero de veces que obtenemos repetido $num5 es: $vecesRep");
?>
```

### Ejercicio 6

La lista de valores de la array es: 1 9 5 1 10 8 4 6 5 3 4 8 4 7 9 6 9 10 9 7 7 4 9 2 9 8 8 7 5 5 1 2 4 4 7 5 5 8 3 4 1 6 6 7 7 1 2 5 7 1  
El numero de veces que obtenemos repetido 1 es: 6.  
La lista de valores tras la criba de la array es: 1,9,5,10,8,4,6,3,7,2  
Tras la criba, el numero de veces que obtenemos repetido 1 es: 1

G. Crea un script que defina una matriz de dos dimensiones, que recoja los datos de "Nombre", "Apellidos", "Idioma" y "Nota Media" para los alumnos de una clase con un total de 5 alumnos. En dicho script define un procedimiento "comprueba", que reciba un número entero (identificador del alumno en la matriz anterior) y recorra la matriz que almacena los datos de dicho alumno, mostrándolos por pantalla. Llámalo desde el cuerpo de la página. Crea otra función "estadísticas", que reciba una matriz y devuelva la media de la clase y el número de aprobados. Llámala desde el cuerpo de la página.

```
<?php
```

```
function comprueba($arrAlumno,$pk){
```

```
/*Buscando en la documentación de PHP, encontramos (
http://php.net/manual/es/function.sizeof.php ) con la siguiente recomendación:
If your array is "huge", it is recommended to set a variable first for this case:
```

```
$max = sizeof($huge_array);
for($i = 0; $i < $max;$i++)
{
code...
}
*/
$tamArr=sizeof($arrAlumno);
for($i=0;$i<$tamArr;$i++){
    if($arrAlumno[0][$i]==$pk){
        echo ("Nombre: ".$arrAlumno[1][$i]."<br>");
        echo ("Apellidos: ".$arrAlumno[2][$i]."<br>");
        echo ("Idioma: ".$arrAlumno[3][$i]."<br>");
        echo ("Nota media: ".$arrAlumno[4][$i]."<br>");
    }
}
```

```
}
function estadisticas($arrAlumno){
    $suma=0;
    $media=0;
    $contadorAprobado=0;
    $tamArr=sizeof($arrAlumno);
    for($i=0;$i<$tamArr;$i++){
        $suma=$suma+$arrAlumno[4][$i];
```

```
        if($arrAlumno[4][$i]>=5){
            $contadorAprobado++;
        }
    }
    $media=$suma/$tamArr;
    echo("La nota media de los alumnos es igual a: ".$media."<br/> El número de aprobados es de: ".$contadorAprobado);
}
```

/\* Para generar una array multidimensional en PHP, hemos consultado: ([https://www.w3schools.com/php/php\\_arrays\\_multi.asp](https://www.w3schools.com/php/php_arrays_multi.asp))

Now the two-dimensional \$cars array contains four arrays, and it has two indices: row and column. To get access to the elements of the \$cars array we must point to the two indices (row and column) Por ello, al recorrer la arrAlumno, hemos seleccionado primero la fila y luego la columna\*/

```
$arrAlumno=array(
    array(1,2,3,4,5),
    array("Miriam", "Manolo", "Miguel", "Marcos","Mercedes"),
    array("Gutiérrez Gutiérrez", "Fernández Fernández", "González González", "Pérez Pérez", "Díaz Díaz"),
    array("Español", "Francés", "Alemán", "Inglés", "Italiano"),
    array(5.4,6.7,5.1,8.9,3.1));
```

```
$pk=2;
```

```
comprueba($arrAlumno, $pk);
estadisticas($arrAlumno);
```

```
?>
```

### Ejercicio 7

Nombre: Manolo  
Apellidos: Fernández Fernández  
Idioma: Francés  
Nota media: 6.7  
La nota media de los alumnos es igual a: 5.84  
El número de aprobados es de: 4

### III. Bibliografía

**GrupoStudium. 2017.** Apuntes de la asignatura. Sevilla : s.n., 2017.

**The PHP Group.** array\_count\_values. [En línea] [Citado el: 25 de 11 de 2017.] <http://php.net/manual/es/function.array-count-values.php>.

—. array\_unique. [En línea] [Citado el: 25 de 11 de 2017.] <http://php.net/manual/es/function.array-unique.php>.

—. sizeof. [En línea] [Citado el: 25 de 11 de 2017.] <http://php.net/manual/es/function.sizeof.php>.

**W3CSchools.com.** PHP array\_count\_values() Function. [En línea] [Citado el: 25 de 11 de 2017.] [https://www.w3schools.com/php/func\\_array\\_count\\_values.asp](https://www.w3schools.com/php/func_array_count_values.asp).