

## E9\_Bibliotecas

### Ejercicio 1. E9\_libreriaListarArray.ph

Código:

```
<html>
<head>
<title>Mostrar Array como una tabla</title>
<style>
table {
border:1px solid black ;
}
th {
border:1px solid black ;
}
td {
border:1px solid black ;
}
</style>
</head>
<body>
<?php
function listarVectorTabla($array) {
echo "Visualiza en forma de tabla:";
echo "<table>";
echo "<th>ELEMENTO</th><th>VALOR</th>";
for ($i = 0; $i < count($array); $i++) {
echo "<tr>";
echo "<td>$i</td><td>$array[$i]</td>";
echo "</tr>";
}
echo "</table>";
}
function listarVectorNoOrdenada($array) {
echo "En forma de lista no ordenada:";
echo "<ul>";
for ($i = 0; $i < count($array); $i++) {
echo "<li>$array[$i]</li>";
}
echo "</ul>";
}
??
</body>
</html>
```

### Ejercicio 2. E9\_libreriaListarArrayPpal.php

Código:

```
<?php
include ("E9_libreriaListarArray.php");
$array = array(10,20,30,40);
echo "Invocamos a la funcion listarVectorTabla...<br><br>";
listarVectorTabla($array);
echo "<br>Invocamos a la funcion listarVectorNoOrdenada...<br><br>";
listarVectorNoOrdenada($array);
?>
```

Salida:

Invocamos a la funcion listarVectorTabla...

Visualiza en forma de tabla:

ELEMENTO	VALOR
0	10
1	20
2	30
3	40

Invocamos a la funcion listarVectorNoOrdenada...

En forma de lista no ordenada:

- 10
- 20
- 30
- 40

## Ejercicio 3. E9\_libreriaArrays.php y E9\_libreriaArraysPpal.php

Código:

```

function vectoresUnordList($vector1, $vector2) {
    echo "Vector de forma abreviada:";
    echo "<ul>";
    for ($i = 0; $i < count($vector1); $i++) {
        echo "<li>Elemento " . $i . " vale: " . $vector1[$i] . "</li>";
    }
    echo "</ul>";
    echo "<br>Vector de forma desarrollada:";
    echo "<ul>";
    for ($i = 0; $i < count($vector2); $i++) {
        echo "<li>Elemento " . $i . " vale: " . $vector2[$i] . "</li>";
    }
    echo "</ul>";
}

function vectoresUnordListRange($vector) {
    echo "Usar función range:";
    echo "<ul>";
    for ($i = 0; $i < count($vector); $i++) {
        echo "<li>Elemento " . $i . " vale: " . $vector[$i] . "</li>";
    }
    echo "</ul>";
}

function vectoresWhile($vector) {
    echo "Contenido del vector de cadenas con WHILE-RESET-NEXT:";
    reset($vector); // Aunque funciona sin poner reset es importante por la coherencia del código
    echo "<ul>";
    while (current($vector)) {
        echo "<li>" . current($vector) . "</li>";
        next($vector);
    }
    echo "</ul>";
}

function vectoresInverso($vector) {
    echo "Contenido del vector de cadenas con WHILE-END-PREV:";
    end($vector);
    echo "<ul>";
    while (current($vector)) {
        echo "<li>" . current($vector) . "</li>";
        prev($vector);
    }
    echo "</ul>";
}

function vectoresForeach($vector) {
    echo "Contenido del vector de cadenas con FOREACH:";
    echo "<ul>";
    foreach ($vector as $nom) {
        echo "<li>" . $nom . "</li>";
    }
}

```

```

<?php
include ("E9_libreriaArrays.php");
$vector = array(1,2,3,4,5,6,7,8,9,10);
$vector2[0] = 10;
$vector2[1] = 20;
$vector2[2] = 30;
$vector2[3] = 40;
$vector2[4] = 50;
$vector2[5] = 60;
$vector2[6] = 70;
$vector2[7] = 80;
$vector2[8] = 90;
$vector2[9] = 100;
$vector3 = range(10, 100, 10);
$vector4 = array("Vicent", "Roberto", "Sergi", "Jordi", "Alvaro");
vectoresUnordList($vector, $vector2);
vectoresUnordListRange($vector3);
vectoresWhile($vector4);
vectoresInverso($vector4);
vectoresForeach($vector4);

```

Salida:

```

Vector de forma abreviada:
<ul>
<li>Elemento 0 vale: 1
<li>Elemento 1 vale: 2
<li>Elemento 2 vale: 3
<li>Elemento 3 vale: 4
<li>Elemento 4 vale: 5
<li>Elemento 5 vale: 6
<li>Elemento 6 vale: 7
<li>Elemento 7 vale: 8
<li>Elemento 8 vale: 9
<li>Elemento 9 vale: 10
</ul>

Vector de forma desarrollada:
<ul>
<li>Elemento 0 vale: 10
<li>Elemento 1 vale: 20
<li>Elemento 2 vale: 30
<li>Elemento 3 vale: 40
<li>Elemento 4 vale: 50
<li>Elemento 5 vale: 60
<li>Elemento 6 vale: 70
<li>Elemento 7 vale: 80
<li>Elemento 8 vale: 90
<li>Elemento 9 vale: 100
</ul>

Usar función range:
<ul>
<li>Elemento 0 vale: 10
<li>Elemento 1 vale: 20
<li>Elemento 2 vale: 30
<li>Elemento 3 vale: 40
<li>Elemento 4 vale: 50
<li>Elemento 5 vale: 60
<li>Elemento 6 vale: 70
<li>Elemento 7 vale: 80
<li>Elemento 8 vale: 90
<li>Elemento 9 vale: 100
</ul>

Contenido del vector de cadenas con WHILE-RESET-NEXT
<ul>
<li>Vicent
<li>Roberto
<li>Sergi
<li>Jordi
<li>Alvaro
</ul>

Contenido del vector de cadenas con WHILE-END-PREV
<ul>
<li>Alvaro
<li>Jordi
<li>Sergi
<li>Roberto
<li>Vicent
</ul>

Contenido del vector de cadenas con FOREACH
<ul>
<li>Vicent
<li>Roberto
<li>Sergi
<li>Jordi
<li>Alvaro
</ul>

```

Ejercicio 4. E9\_libreriaArraySuma.php y E9\_libreriaArraySumaPpal.php

Código:

```
<?php
function arraySuma($vector) {
    $suma = 0;
    echo "<b>Declarar y Recorrer Vectores</b><br><br>";
    echo "Declara array unidimensional<br><br>";
    echo "Recorre array<br><br>";
    echo "Número de elementos del array: " . count($vector) . "<br><br>";
    echo "Los elementos del array son:<br>";
    foreach ($vector as $num) {
        echo $num . "<br>";
        $suma += $num;
    }
    echo "<br><br><b>SUMA " . $suma . "</b>";
}

function arraySumaTabla($vector) {
    $suma = 0;
    echo "<table>";
    echo "<tr><th>Posición</th>";
    echo "<th>Valor</th></tr>";
    for($i = 0; $i < count($vector); $i++) {
        echo "<tr><td>" . $i . "</td>";
        echo "<td>" . $vector[$i] . "</td></tr>";
        $suma += $vector[$i];
    }
    echo "<tr><th>SUMA</th>";
    echo "<td>" . $suma . "</td></tr>";
    echo "</table>";
}

?>
```

```
<?php
include ("E9_libreriaArraySuma.php");
$vector = range(100, 200, 25);
echo "Invocando a arraySuma<br><br>";
arraySuma($vector);
echo "<br><br>Invocando a arraySumaTabla<br><br>";
arraySumaTabla($vector);

?>
```

Salida:

Invocando a arraySuma

**Declarar y Recorrer Vectores**

Declara array unidimensional

Recorre array

Número de elementos del array: 5

Los elementos del array son:

100  
125  
150  
175  
200

**SUMA 750**

Invocando a arraySumaTabla

Posición	Valor
0	100
1	125
2	150
3	175
4	200
SUMA	750

## Ejercicio 5. E9\_libreriaMediaAritmetica.php

Código:

```
<?php
function mediaValores() {
    $nArgs = func_num_args();
    $lista = func_get_args();
    $suma = 0;

    echo "<br><b>Función Media Aritmética</b><br>";
    echo "Número de argumentos utilizados: " . $nArgs . "<br>";
    echo "<br>Valor de los argumentos utilizados:<br>=====<br>";

    for ($i = 0; $i < $nArgs; $i++) {
        $n = $i + 1;
        echo "Parámetro " . $n . " ==> valor: " . $lista[$i] . "<br>";
        $suma += $lista[$i];
    }

    $media = $suma / $nArgs;
    echo "<br>La media de dichos argumentos es: $media</b>";
}

function mediaArray($lista) {
    $nArgs = count($lista);
    $suma = 0;

    echo "<br><b>Función Media Aritmética con array</b><br>";
    echo "Número de argumentos utilizados: " . $nArgs . "<br>";
    echo "<br>Valor de los argumentos utilizados:<br>=====<br>";

    for ($i = 0; $i < $nArgs; $i++) {
        $suma += $lista[$i];
    }

    print_r($lista);

    $media = $suma / $nArgs;
    echo "<br>La media de dichos argumentos es: $media</b>";
}

function mediaValoresRefer($num1, $num2) {
    echo "<b>Función Media Aritmética</b><br>";
    echo "Número de argumentos utilizados: 2<br><br>";
    echo "Valor de los argumentos utilizados: <br>$num1 y $num2";
    $sum2 = ($num1 + $num2) / 2;
    echo "<br>No hace falta hacer return<br>";
}
?>
```

## Ejercicio 6. E9\_libreriaMediaAritmeticaPpal.php

Código:

```
<?php
include ("E9_libreriaMediaAritmetica.php");
$lista = array(10,20,30,40,50,60,70);
$a = 10;
$b = 40;
echo "Realizando llamada a mediaValores...<br><br>";
mediaValores(10,20,30,40);
echo "Realizando llamada a mediaArray...<br><br>";
mediaArray($lista);
echo "Realizando llamada a mediaValoresRefer...<br><br>";
mediaValoresRefer($a, $b);
echo "<br>La media es: $b";
?>
```

Salida:

Realizando llamada a mediaValores...

**Función Media Aritmética**  
 Numero de argumentos utilizados: 4

Valor de los argumentos utilizados  
 =====

Parámetro 1 ==> valor: 10  
 Parámetro 2 ==> valor: 20  
 Parámetro 3 ==> valor: 30  
 Parámetro 4 ==> valor: 40

**La media de dichos argumentos es: 25**

Realizando llamada a mediaArray...

**Función Media Aritmética con array**  
 Numero de argumentos utilizados: 7

Valor de los argumentos utilizados  
 =====

Array ( [0] => 10 [1] => 20 [2] => 30 [3] => 40 [4] => 50 [5] => 60 [6] => 70 )

**La media de dichos argumentos es: 40**

Realizando llamada a mediaValoresRefer...

**Función Media Aritmética**  
 Número de argumentos utilizados: 2

Valor de los argumentos utilizados:  
 10 y 40  
 No hace falta hacer return

La media es: 25

## Ejercicio 7. E9\_Funcionproducto.php y E9\_invocaFuncionProducto.ph

Código:

```
<?php
function producto($a,$b) {
    echo "<br>Se encuentra en el archivo <b>funcionProducto.php</b>";
    return $a * $b;
}

?>
```

---

```
<?php
include ("E9_Funcionproducto.php");
echo "Asignamos valores a las variables:<br><br>";
$a = 10;
$b = 20;
echo "multiplicando: $a <br> multiplicador: $b<br>";
echo "<br>Invocamos a la función";
$res = producto($a, $b);
echo "<br>El resultado de $a x $b es <b>$res</b>";

?>
```

Salida:

Asignamos valores a las variables:

multiplicando: 10  
multiplicador: 20

Invocamos a la función

Se encuentra en el archivo **funcionProducto.php**  
El resultado de 10 x 20 es **200**

## Ejercicio 8. E9\_funcionProductoVariosValores.php

Código:

```
<?php
function productoVariosValores(){
    $nArgs = func_num_args();
    $lista = func_get_args();
    $prod = 1;

    echo "<br><b>Función Producto</b><br>";
    echo "Numero de elementos a multiplicar: " . $nArgs . "<br>";
    echo "<br>Valores<br>=====<br>";

    for ($i = 0; $i < $nArgs; $i++){
        $n = $i + 1;
        echo "Parámetro " . $n . " ==> valor: " . $lista[$i] . "<br>";
        $prod *= $lista[$i];
    }
    echo "<h2>El producto es: $prod</h2>";

}

?>
```

## Ejercicio 9. E9\_funcionSumaProducto.php y E9\_ppalSumaProducto.ph

Código:

```

<?php
    function producto($a,$b) {
        return $a * $b;
    }
    function suma($a,$b) {
        return $a + $b;
    }
?>

<?php
    include ("E9_funcionSumaProducto.php");
    echo "Asignamos valores a los dos valores<br>";
    $a = 10;
    $b = 20;
    echo '$a =' . $a . "<br>". '$b =' . $b;
    echo "<br><br>Llamada a Funcion Suma";
    $suma = suma(10,20);
    echo "<br>La suma de $a y $b es $suma";
    echo "<br><br>Llamada a Funcion Producto";
    $prod = producto(10,20);
    echo "<br>El producto de $a y $b es $prod"
?>

```

Salida:

Asignamos valores a los dos valores  
 \$a =10  
 \$b =20

Llamada a Funcion Suma  
 La suma de 10 y 20 es 30

Llamada a Funcion Producto  
 El producto de 10 y 20 es 200

## Ejercicio 10. E9\_libreriaProductos.php

Código:

```

<?php
include ("E9_funcionProducto.php");
include ("E9_funcionProductoVariosValores.php");
?>

```

## Ejercicio 11. E9\_libreriaProductosPpal.php

Código:

```
<?php
include ("E9_libreriaProductos.php");
echo "Estamos en Principal<br>";
$a = 10;
$b = 20;
$prod = producto($a, $b);
echo "<br>Producto de $a y $b = $prod<br>";
productoVariosValores(2,3,4,6);
?>
```

Salida:

Estamos en Principal

Se encuentra en el archivo **funcionProducto.php**  
Producto de 10 y 20 = 200

**Función Producto**

Numero de elementos a multiplicar: 4

Valores

---

---

Parámetro 1 ==> valor: 2  
Parámetro 2 ==> valor: 3  
Parámetro 3 ==> valor: 4  
Parámetro 4 ==> valor: 6

**El producto es: 144**