

# PROGRAMACIÓN II



**SEMANA 05**  
**ARREGLOS**

Eric Gustavo Coronel Castillo  
[ecoronel@uch.edu.pe](mailto:ecoronel@uch.edu.pe)  
[gcoronelc.blogspot.com](http://gcoronelc.blogspot.com)

# ARREGLOS

El uso de variables es la forma más simple de guardar datos en memoria durante la ejecución de un programa, pero resulta inadecuado para algunos procesos, por eso debemos recurrir a una estructura de datos que permita almacenar varias variables como una sola unidad, para luego procesarlas mediante bucles.

## Índice

1	¿QUÉ ES UN ARREGLO? .....	4
2	ARREGLOS UNIDIMENSIONALES .....	6
3	ARREGLOS MULTIDIMENSIONALES .....	11

## 1 ¿Qué es un Arreglo?

---

Arreglo (Array en inglés) es una estructura de datos conformada por un conjunto de variables, agrupadas bajo un mismo nombre, a las cuales accedemos mediante un índice.

\$lista	
0	"Impresora"
1	100
2	550.00

**Figura 1** Ejemplo de un arreglo.

El primer elemento del arreglo inicia con índice 0, y los elementos que conforman el arreglo pueden ser de cualquier tipo.

### Ejemplo 1

Este ejemplo crea un arreglo, luego muestra cada uno de sus elementos y su respectivo tipo de dato.

Archivo: prog2\sem05\ejm01.php

---

```
<?php
// Creación del arreglo
$lista[] = "Impresora";
$lista[] = 100;
$lista[] = 165.79;
?>

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>Capítulo 06 - Ejemplo 01</title>
  </head>
  <body bgcolor="#D2EBF7">
    <h3>Arreglos</h3>
    <?php
      echo( "\$lista[0] = " . $lista[0] . "<br>" );
      echo( "\$lista[1] = " . $lista[1] . "<br>" );
      echo( "\$lista[2] = " . $lista[2] . "<br>" );
    </?php>
  </body>
</html>
```

```
echo( "<br>" );  
echo( "\\$lista[0] es de tipo " . gettype($lista[0]) . "<br>" );  
echo( "\\$lista[1] es de tipo " . gettype($lista[1]) . "<br>" );  
echo( "\\$lista[2] es de tipo " . gettype($lista[2]) . "<br>" );  
?>  
</body>  
</html>
```

Resultado:

## Arreglos

**\$lista[0] = Impresora**

**\$lista[1] = 100**

**\$lista[2] = 165.79**

**\$lista[0] es de tipo string**

**\$lista[1] es de tipo integer**

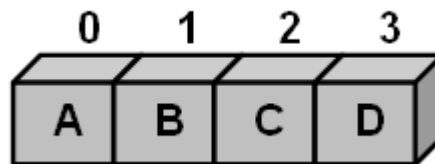
**\$lista[2] es de tipo double**

Como podemos apreciar, para acceder a un elemento utilizamos un índice que inicia en cero (0), en este caso el primer elemento corresponde a una variable de tipo *string*, el segundo elemento a una variable de tipo *integer*, y finalmente el tercer elemento a una variable de tipo *double*.

## 2 Arreglos Unidimensionales

---

Un arreglo unidimensional es una lista de valores a los cuales accedemos mediante un índice que inicia en cero.



**Figura 2** Representación gráfica de un arreglo unidimensional.

Para crear un arreglo tenemos varias alternativas, por ejemplo para iniciar la creación de una lista de frutas, la instrucción sería:

```
$lista[] = "Pera";
```

En este caso el índice de este primer elemento es cero (0), para agregar otro elemento utilizamos una instrucción similar:

```
$lista[] = "Naranja";
```

Para saber el tamaño de un arreglo podemos utilizar la función `count()`, tal como se ilustra a continuación:

```
$n = count($lista);
```

### Ejemplo 2

El siguiente programa ilustra el uso de la función `count()` y cómo podemos hacer un recorrido por los elementos de un arreglo.

**Archivo:** prog2\sem05\ejm02.php

---

```
<?php
// Creación del Arreglo
$lista[] = "Pera";
$lista[] = "Naranja";
$lista[] = "Uva";
```

```
$lista[] = "Manzana";  
?>  
  
<html>  
  <head>  
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />  
    <title>Capítulo 06 - Ejemplo 02</title>  
  </head>  
  <body bgcolor="#D2EBF7">  
    <h3>Arreglos</h3>  
    <?php  
      $n = count($lista);  
      for( $j=0; $j < $n; $j++ ) {  
        echo( "\$lista[$j] = $lista[$j] <br>" );  
      }  
    ?>  
  </body>  
</html>
```

El resultado:

### Arreglos

```
$lista[0] = Pera  
$lista[1] = Naranja  
$lista[2] = Uva  
$lista[3] = Manzana
```

También podemos utilizar la función `array()` para crear un arreglo, tal como se ilustra a continuación:

```
$lista = array("Chiclayo","Trujillo","Piura","Tumbes");
```

### Ejemplo 3

Este ejemplo crea un arreglo utilizando la función `array()`, luego lista todos sus elementos.

Archivo: Prog2\sem05\ejm03.php

```
<?php
// Creación del Arreglo
$lista = array("Chiclayo","Trujillo","Piura","Tumbes");
?>

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>Capítulo 06 - Ejemplo 03</title>
  </head>
  <body bgcolor="#D2EBF7">
    <h3>Conoce el Norte de Perú</h3>
    <?php
      $n = count($lista);
      for( $j=0; $j < $n; $j++ ) {
        echo "\$lista[$j] = $lista[$j] <br>";
      }
    ?>
  </body>
</html>
```

El resultado:

### Conoce el Norte de Perú

```
$lista[0] = Chiclayo
$list[1] = Trujillo
$list[2] = Piura
$list[3] = Tumbes
```

Para recorrer los elementos de un arreglo podemos utilizar un bucle haciendo referencia a su índice, pero necesitamos conocer el tamaño del arreglo. La estructura de control `foreach` nos facilita el recorrido por los elementos de un arreglo por qué no necesitamos conocer su tamaño, a continuación se ilustra su sintaxis:

```
foreach( $arreglo as $elemento) {  
  
    instrucciones;  
  
}
```



## Ejemplo 4

En el siguiente ejemplo se ilustra el uso de la estructura `foreach`.

Archivo: prog2\sem05\ejm04.php

```
<?php
$lista = array("Arroz con Pato","Ceviche","Seco de Cabrito",
    "Seco de Chavelo","Sudado de Mero", "Espesado");
?>

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>Capítulo 06 - Ejemplo 04</title>
  </head>
  <body bgcolor="#D2EBF7">
    <h3>Gastronomía Del Norte de Perú</h3>
    <ol>
      <?php
      foreach( $lista as $dato ) {
        echo( "<li>$dato</li>" );
      }
      ?>
    </ol>
  </body>
</html>
```

Resultado:

### Gastronomía Del Norte de Perú

1. Arroz con Pato
2. Ceviche
3. Seco de Cabrito
4. Seco de Chavelo
5. Sudado de Mero
6. Espesado

### 3 Arreglos Multidimensionales

Si cada elemento de un arreglo unidimensional es otro arreglo, entonces tenemos un arreglo de dos dimensiones, si los elementos de este segundo arreglo son otros arreglos, entonces tenemos un arreglo de tres dimensiones, y así sucesivamente.

Cuando se trata de dos dimensiones cada elemento en particular está identificado por dos índices, si fuese de tres dimensiones serían tres índices y así sucesivamente.

Veamos la siguiente asignación:

```
$lista = array(
    array(45,67,80,45),
    array(67,89,23,67),
    array(12,87,72,54)
);
```

Estamos creando un arreglo de dos dimensiones, de 3 filas por 4 columnas, para acceder a un elemento se debe usar la siguiente sintaxis:

```
$lista[filas][columna]
```

	0	1	2	3
0	45	67	80	45
1	67	89	23	67
2	12	87	72	54

**Figura 3** Representación gráfica de un arreglo de dos dimensiones.

Si necesitamos hacer un recorrido por todo el arreglo deberíamos utilizar la siguiente plantilla:

```
for( $f = 0; $f < count( $lista); $f++ ) {

    for( $c = 0; $c < count( $lista[ $f ] ); $c++ ) {

        // Proceso a desarrollar, el acceso al elemento es así: $lista[$f][$c]
```

}

}

## Ejemplo 5

En este ejemplo se ilustra el recorrido de un arreglo de dos dimensiones, conocido también como matriz.

Archivo: prog2\sem05\ejm05.php

```
<?php
// Creando el Arreglo
$lista = array(
    array("Perú","Lima"),
    array("Bolivia","La Paz"),
    array("Brazil","Brasilia"),
    array("Argentina","Buenos Aires"),
    array("Colombia","Bogotá")
);
?>

<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>Capítulo 06 - Ejemplo 05</title>
</head>
<body bgcolor="#D2EBF7">
    <h3>Conoce América Latina</h3>
    <table width="300" border="1">
        <tr>
            <th width="128">País</th>
            <th width="156">Capital</th>
        </tr>
        <?php for( $f = 0; $f < count($lista); $f++ ) { ?>
            <tr>
                <td><?php echo($lista[$f][0]) ?></td>
                <td><?php echo($lista[$f][1]) ?></td>
            </tr>
        <?php } ?>
    </table>
</body>
</html>
```

El `for` hace el recorrido por todas las filas; como son solamente dos columnas, se accede directamente utilizando el índice 0 para la primera columna y el índice 1 para la segunda columna.

El resultado es el siguiente:

### Conoce América Latina

País	Capital
Perú	Lima
Bolivia	La Paz
Brazil	Brasilia
Argentina	Buenos Aires
Colombia	Bogotá