

PROGRAMACIÓN EN PHP

Programando con PHP

SINTAXIS BÁSICA: VARIABLES

VARIABLES: ¿QUÉ SON?

- ✖ Una variable es un espacio de memoria al que se le da nombre y puede almacenar un determinado valor.
- ✖ El valor de las variables puede ser modificado a lo largo del tiempo.
- ✖ Para acceder al valor de la variable sólo hay que hacer referencia a su nombre.

VARIABLES

- ✖ En PHP las variables se crean en el instante que son utilizadas por primera vez.
- ✖ Para inicializarlas se utiliza el operador de asignación (=).
- ✖ A partir de ese momento podemos utilizarlas para cualquier cosa.

VARIABLES

- ✗ PHP es un lenguaje débilmente tipado:
 - + una variable podrá almacenar cualquier tipo de información.
- ✗ En PHP todas las variables van precedidas del símbolo \$, seguido de una letra o un (_)

Inválidos	Motivo	Válidos
\$Valor actual	Contiene un espacio	\$Valor_actual
\$#Datos	Contiene carácter no válido	\$NumeroDeDatos
\$2Saldcs	Empieza por número	\$N
\$Prueba,valor	Contiene carácter no válido	\$n

VARIABLES

- ✖ Hay que tener en cuenta que PHP distingue entre mayúsculas y minúsculas.
- + No es la misma variable \$n que \$N

VARIABLES: EJERCICIOS

- ✗ Crear un documento PHP en el que se cree una variable llamada “numero”. Asignar a esa variable el valor 15 y mostrarlo por pantalla.

+ Utilizar

- ✗ Color de fondo el color ‘9922AA’
- ✗ Texto tamaño 13 y color blanco

```
<?php
    echo "<b><font color='white' size='13'>";
    $numero = 15;
    echo $numero;
    echo "</font></b>";
?>
```

VARIABLES: EJERCICIOS

- ✖ Crear una página Web en PHP en la que haya una única variable. Esta variable deberá contener los siguientes valores:
 - + 3
 - + Casa
 - + 3,254
 - + Hola mundo
- ✖ Mostrar dichos valores en líneas independientes


```
1 <html>
2   <head>
3     <title> PHP </title>
4   </head>
5   <body>
6     <?php
7         $variable = 3;
8         echo "variable vale: $variable";
9         echo "<br>";
10        $variable = "casa";
11        echo "variable vale: $variable";
12        echo "<br>";
13        $variable = 3.254;
14        echo "variable vale: $variable";
15        echo "<br>";
16        $variable = "hola mundo";
17        echo "variable vale: $variable";
18        echo "<br>";
19    ?>
20  </font>
21 </body>
22 </html>
```

VARIABLES: EJERCICIOS

- ✖ Escribir un programa que nos pregunte nuestro nombre y nos salude.
- ✖ *NOTA: por ahora no podemos pedir al usuario que nos dé un dato: simularemos la petición de datos asignando directamente el valor a una variable.*

```
<?php
    $nombre='Pepe';
    echo "Hola $nombre";
?>
```

```
<?php
    $nombre='Pepe';
    echo "Hola".$nombre;
?>
```

VARIABLES: EJERCICIOS

- ✖ Escribir un programa que nos pregunte nuestro nombre y nos salude dándonos la bienvenida a nuestra web.

```
<?php
    $nombre = 'Pepe';
    echo "Hola $nombre bienvenido a nuestra website";
?>
```

```
<?php
    $nombre = 'Pepe';
    echo "Hola ".$nombre." bienvenido a nuestra Website";
?>
```

VARIABLES: EJERCICIOS

- ✗ Escribir un programa que nos pida insertar una palabra. El programa deberá mostrar por pantalla la palabra subrayada.

```
<?php
    $palabra = 'margarita';
    echo "La palabra insertada es <u>$palabra</u>";
?>
```

```
<?php
    $palabra = 'margarita';
    echo "La palabra insertada es <u>".$palabra."</u>";
?>
```


Programando con PHP

SINTAXIS BÁSICA: TIPOS DE DATOS

TIPOS DE DATOS

× Numéricos

- + Permite almacenar números que van desde los -2 billones hasta 2 billones.
- + Para almacenar en una variable un valor numérico se asigna el valor utilizando el operador de asignación (=).

TIPOS DE DATOS

✗ Cadenas

- + Representan a un conjunto de 0 o más caracteres.
- + Se encierran entre (“) o (‘).

✗ Con “:

- + Podremos incluir en la cadena nombres de variables (en ese caso se mostrará el valor de la variable).
- + Podremos incluir en la cadena ‘.

✗ Con ‘:

- + No se evaluará nada de lo que aparezca en la cadena.

<H2>Trabajando Con Caderas de caracteres </H2>

<?php

```
$lenguaje = "PHP";  
$ver = "v5";  
echo "<B>Estamos 'trabajando' con $lenguaje $ver"  
echo "</B><BR><BR>";  
echo 'La variable $lenguaje contiene: ';  
echo "$lenguaje <BR>";  
echo 'La variable $ver contiene: ';  
echo $ver;
```

?>

Trabajando Con Caderas de caracteres

Estamos 'trabajando' con PHP v5

La variable \$lenguaje contiene: PHP

La variable \$ver contiene: v5

<h2> Comillas dobles y simples </h2>

<?php

```
$cadena=' un tesoro ';  
echo "la cadena '$cadena' contiene: $cadena"  
echo "<br><br>";  
echo "La cadena \"$cadena\" contiene: $cadena"  
echo "<br><br>";  
echo 'La cadena "$cadena" contiene: $cadena'  
echo '<br><br>';
```

?>

Comillas dobles y simples

la cadena ' un tesoro ' contiene: un tesoro

La cadena " un tesoro " contiene: un tesoro

La cadena "\$cadena" contiene: \$cadena

Programando con PHP

SINTAXIS BÁSICA: OPERADORES

OPERADORES ARITMÉTICOS

Operador	Ejemplo	Descripción
+	$\$a + \b	Suma dos operandos
-	$\$a - \b	Resta dos operandos
*	$\$a * \b	Multiplica dos operandos
/	$\$a / \b	Divide dos operandos
%	$\$a \% \b	Resto de la división entera

OPERADORES DE ASIGNACIÓN

Operador	Ejemplo	Descripción
=	$\$a = \b	$\$a$ toma el valor de $\$b$
+=	$\$a += \b	Equivale a $\$a = \$a + \$b$
-=	$\$a -= \b	Equivale a $\$a = \$a - \$b$
/=	$\$a /= \b	Equivale a $\$a = \$a / \$b$
*=	$\$a *= \b	Equivale a $\$a = \$a * \$b$
%=	$\$a \% = \b	Equivale a $\$a = \$a \% \$b$
.=	$\$a .= \b	Equivale a $\$a = \$a . \$b$

OPERADORES DE INCREMENTO Y DECREMENTO

Operador	Ejemplo	descripción
++	++\$a	Preincremento: incrementa \$a en uno y después devuelve \$a
	\$a++	Postincremento: devuelve \$a y después incrementa en uno \$a
--	--\$a	Predecremento: decrementa en uno \$a y después devuelve \$a
	\$a--	Postdecremento: devuelve \$a y después decrementa en uno \$a

OPERADORES DE COMPARACIÓN

Operador	Ejemplo	Devuelve TRUE cuando
==	\$a == \$b	Los operandos son iguales
!=	\$a != \$b	Los operandos son distintos.
===	\$a === \$b	Los operandos son idénticos: iguales y del mismo tipo.
!==	\$a !== \$b	Los operandos no son iguales o del mismo tipo.
<	\$a < \$b	El operando de la izquierda es menor que el de la derecha
>	\$a > \$b	El operando de la izquierda es mayor que el de la derecha
<=	\$a <= \$b	El operando de la izquierda es menor o igual que el de la derecha
>=	\$a >= \$b	El operando de la izquierda es mayor o igual que el de la derecha

OPERADORES LÓGICOS

Operador	Ejemplo	Devuelve TRUE cuando
&&	\$a && \$b	\$a y \$b son ambos true
And	\$a and \$b	
	\$a \$b	\$a o \$b son true
Or	\$a or \$b	
!	!\$a	\$a es false, niega el valor lógico de la variable
xor	\$a xor \$b	\$a es true o \$b es true, pero no lo son los dos a la vez

VARIABLES: EJERCICIOS

- ✖ Crear una pagina Web en la que se creen 2 variables una llamada numero1 y otra llamada numero2.
 - + En la variable numero1 meteremos el valor 3
 - + En la variable numero2 meteremos el valor 4
 - + En la página Web se mostrarán los valores de ambas variables y el resultado de sumarlas.


```
1 <html>
2   <head>
3     <title> PHP </title>
4   </head>
5 <body>
6   <?php
7       $numero1 = 3;
8       $numero2 = 4;
9       echo "numero 1 vale: $numero1";
10      echo "<br>";
11      echo "numero 2 vale: $numero2";
12      echo "<br>";
13      $suma = $numero1 + $numero2;
14      echo "la suma de los dos valores es: $suma";
15
16   ?>
17 </body>
18 </html>
```

VARIABLES: EJERCICIOS

- ✖ Se necesita obtener la nota media de un estudiante a partir de sus tres notas parciales.

```
<?php
```

```
    $nota1 = 7;
```

```
    $nota2 = 5;
```

```
    $nota3 = 9;
```

```
    $notaMedia = ($nota1 + $nota2 + $nota3)/3;
```

```
    echo "El resultado de la nota media es: $notaMedia";
```

```
?>
```

EJERCICIOS

- ✗ Crear una página Web en PHP que tome dos números y muestre la suma de ellos.
 - + Se deberán mostrar los valores de los dos números y el de la suma
- ✗ Crear una Web en PHP que con dos números cuales quiera muestre una lista de la siguiente forma:
 - Suma -> $25+13 = 38$
 - Resta -> $25-13 = 12$
 - Multiplicación -> $25*13 = 325$
 - División -> $25/13 = 1.9230$

EJERCICIOS

✗ Crear un documento PHP que muestre como salida la tabla de multiplicar del 5.

+ Se deberá:

✗ Hacer dentro de una lista como la siguiente:

- $5 \times 1 = 5$
- $5 \times 2 = 10$
- $5 \times 3 = 15$
- $5 \times 4 = 20$
- $5 \times 5 = 25$
- ...
- $5 \times 10 = 50$

✗ Utilizar una única variable

Programando con PHP

LOS ARRAYS

ARRAYS

- ✖ Conjunto de datos almacenados bajo el mismo nombre.
- ✖ Son posiciones de memoria consecutivas, todas ellas referenciadas por el mismo nombre.
- ✖ Un array en PHP no tiene un tamaño predefinido, con lo que puede tener tantos valores como sea necesario.

ARRAYS

- ✖ Podemos acceder a cada uno de los elementos del array haciendo referencia a la posición que ocupan dentro del conjunto de valores.
- ✖ Las posiciones de un array siempre empiezan por 0.

Posición	0	1	2	3
Valor	25	36	21	58

ARRAYS

- ✖ Puesto que PHP es un lenguaje débilmente tipado, los arrays pueden estar compuestos por elementos de diferente tipo.

Posición	0	1	2	3	4
Valor	León	Seat		2.500	V6

ARRAYS

- ✖ La declaración de un array posicional se hace como si cada posición fuera una variable independiente.
- ✖ Para indicar a qué posición queremos hacer referencia utilizamos los corchetes ([])

```
7 <tr>
8     <td> Posición </td>
9     <td> 0 </td>
10    <td> 1 </td>
11    <td> 2 </td>
12    <td> 3 </td>
13 </tr>
```

```
14 <tr>
15     <td> Valor </td>
```

```
16 <?php
```

```
17     $array[0]=25;
```

```
18     $array[1]=36;
```

```
19     $array[2]=21;
```

```
20     $array[3]=58;
```

```
21     echo "<td> $array[0] </td>";
```

```
22     echo "<td> $array[1] </td>";
```

```
23     echo "<td> $array[2] </td>";
```

```
24     echo "<td> $array[3] </td>";
```

```
25     ?>
```

```
26 </tr>
```


ARRAYS

- ✖ Para asignar un valor a la última posición de un array no hace falta indicar nada entre los corchetes.

```
<?php
    $matriz1[0]="rojo";
    $matriz1[1]="azul";
    $matriz1[2]="violeta";
    $matriz1[3]="";
    $matriz1[]="rosa";
    $matriz1[]="amarillo";
```

```
?>
```

EJERCICIOS

- ✖ Crear un array de 4 posiciones y mostrar los valores de la siguiente forma:
 - + En la posición 1 hay un: *número*
 - + En la posición 2 hay un: *número*
 - + En la posición 3 hay un: *número*
 - + En la posición 4 hay un: *número*
- ✖ Crear un array de 3 posiciones, asignarle a cada posición un valor. Después mostrar por pantalla el siguiente mensaje:
 - + *Valor1 + valor2 + valor3 = suma*

EJERCICIOS

- ✗ Crear un array con los nombre de los alumnos de una clase (al menos 5 alumnos) y mostrarlos de la siguiente forma:
 - + Los alumnos de la clase son:
 - ✗ *Nombre1*
 - ✗ *Nombre2*
 - ✗ *Nombre3*
 - ✗ *Nombre4*

EJERCICIOS

- ✗ Crear un array en el que estén los datos personales de alguien (Nombre, apellidos, edad y DNI) y mostrarlos de la siguiente forma:
 - + Los datos personales de *Nombre* son:
 - ✗ Apellidos: *apellidos*
 - ✗ Edad: *edad*
 - ✗ DNI: *DNI*

EJERCICIOS

- ✖ Crear el siguiente array posicional y mostrarlo dentro de una tabla igual a la siguiente:

0	1	2	3	4	5	6
Jorge	Pérez	35	1.77	80	Moreno	Soltero

EJERCICIOS

- ✗ Crear un array que almacene los días de la semana y mostrarlos dentro de una lista.
- ✗ Crear un array que almacene los siguientes valores:
 - + 8
 - + Hola
 - + 25.6
 - + Adios
- ✗ Mostrar el array en orden inverso.

MATRICES

- ✖ Son arrays bidimensionales, es decir, arrays que se organizan en filas y columnas.
- ✖ Para acceder a cada posición de una matriz tendremos que referenciarla con 2 índices: fila y columna en la que está situado.
- ✖ `$matriz[1][2]=5`
- ✖ `$matriz[2][2]=9`
- ✖ `$matriz[1][3]=43`

2	85	32	10
68	23	5	43
25	12	9	0

EJERCICIOS

- ✖ Crear una matriz de 3x5 que contenga 15 números y mostrar:
 - + Sólo los pertenecientes a la 3 fila.
 - + Sólo los pertenecientes a la 1 columna.
- ✖ Crear una matriz de 4x3 en la que se van a almacenar el nombre, apellido y edad de 4 personas. Mostrarlo después como la siguiente tabla:

Pepe	Lopez	35
Maria	Fernandez	15
Paloma	Ruiz	68
Pedro	Molina	24

EJERCICIOS

- ✗ Crear una matriz de 3 filas y 2 columnas, mostrar después la siguiente tabla.

Numero1	4	8
Numero2	4	6
Numero3	2	9
Suma	10	23

- ✗ Crear una matriz de 5x5 y rellenarla con letras. Después, pedir al usuario qué casilla quiere visualizar y mostrar el siguiente mensaje:
 - + En la casilla *fila*x*columna* está la letra: *letra*

Programando con PHP

LAS CONSTANTES

CONSTANTES

- ✖ Una constante es un tipo especial de identificador.
- ✖ Admite un valor único que no podrá cambiar durante la ejecución del programa.
- ✖ Permite mantener un valor fijo bajo un nombre sin tener el riesgo de que se modifique durante la ejecución de el programa.

CONSTANTES

- ✖ Crear una constante en PHP es tan fácil como usar una de las siguientes funciones:
 - + Define(nombre_constante, valor): crea la constante con el valor.
- ✖ Para hacer referencia a una constante no hay que poner \$ delante de su nombre.

EJERCICIOS

- ✖ Crear un documento PHP que contenga una constante de nombre PRUEBA. Al crearla asignarle un valor numérico cualquiera. Comprobar cómo al intentar cambiar el valor PHP nos devuelve un error.

CONSTANTES PREDEFINIDAS

Constante	Significado
PHP_VERSION	Cadena que representa la versión del intérprete de PHP en uso.
PHP_OS	Cadena con el nombre del sistema operativo en el que se está ejecutando el intérprete de PHP
TRUE	Verdadero
FALSE	Falso
E_ERROR	Información sobre errores distintos a los de interpretación del cual no es posible recuperarse.
E_PARSE	Informa que el intérprete encontró una sintaxis inválida en el archivo de comandos. Finaliza la ejecución.
E_NOTICE	Informa que se produjo algo incorrecto que puede provenir o no de un error. A ejecución continúa.
E_WARNING	Denota un error que no impide que continúe la ejecución.
E_ALL	Conjunto con todos los errores que se han producido.

PROGRAMACIÓN EN PHP
