



APRENDERAPROGRAMAR.COM

TIPOS DE VARIABLES EN  
PHP. DECLARACIÓN Y  
ASIGNACIÓN. LA  
INSTRUCCIÓN ECHO PARA  
INSERTAR TEXTO O  
CÓDIGO. (CU00816B)

Sección: Cursos

Categoría: Tutorial básico del programador web: PHP desde cero

Fecha revisión: 2029

**Resumen:** Entrega nº16 del Tutorial básico "PHP desde cero".

Autor: Enrique González Gutiérrez

## VARIABLES EN PHP

Las variables en PHP son representadas con un signo de dólar (\$) seguido por el nombre de la variable. El nombre de la variable es sensible a minúsculas y mayúsculas. Es decir, las siguientes declaraciones de variables son distintas entre ellas:



`$variable`, `$Variable`, `$variAble`, `$VariAble`, etc. representan a distintas variables.

Recordar que una variable es un espacio de la memoria del ordenador (en este caso del servidor) a la que asignamos un contenido que a grandes rasgos puede ser un valor numérico (sólo números, con su valor de cálculo) o alfanumérico (sólo texto o texto con números, sin valor de cálculo).

Para declarar un nombre de variable válido, éste tiene que empezar con una letra (o si no empieza con una letra, tendrá que hacerlo con un carácter de subrayado), seguido de cualquier número de letras, números y caracteres de subrayado, sin espacios. Veamos algunos ejemplos:

| Posible nombre de variable                  | Validez                      |
|---|------------------------------|
| <code>\$4variable</code>                    | Nombre de variable inválido. |
| <code>\$_4variable</code>                   | Nombre de variable válido.   |
| <code>\$variable4</code>                    | Nombre de variable válido.   |
| <code>\$otra</code>                         | Nombre de variable válido.   |
| <code>\$1_otra</code>                       | Nombre de variable inválido. |
| <code>\$variable_de_nombre_muy_largo</code> | Nombre de variable válido.   |
| <code>\$ABC</code>                          | Nombre de variable válido.   |
| <code>\$ A B C</code>                       | Nombre de variable inválido. |
| <code>\$A_y_B_x_C</code>                    | Nombre de variable válido.   |

Cuando el intérprete PHP detecta un error en la sintaxis al declarar una variable, cuando tratamos de acceder a la página php nos aparecerá un mensaje de error. Por ejemplo si en un archivo php incluimos una línea como `$ A B C = 5`; donde el nombre de variable es inválido ya que tiene espacios intermedios, nos aparecerá un mensaje de error:

Parse error: syntax error, unexpected T\_STRING, expecting T\_VARIABLE or '\$' in /home/miraiz/domains/midominio.com/public\_html/prueba.php on line 3. Este mensaje nos informa de que hay un problema en el archivo php y nos indica la línea donde el intérprete considera que está el problema.

## DIFERENCIAS ENTRE LOS OPERADORES DE ASIGNACIÓN E IGUALDAD

### Asignación (=)

El operador de asignación es "=". Se podría pensar que es como un "igual a". No lo es. El operador igual en programación no tiene el mismo sentido que en matemáticas. En programación, significa que el valor de la expresión de la derecha se establece en el operando de la izquierda.

Fijate en este ejemplo y en los comentarios.

```
<?php
    $a = 3; // asignamos el valor 3 a la variable $a
    $b = "cadena"; // asignamos el valor "cadena" a la variable $b
?>
```

### Igualdad (==)

El comparador de igualdad es ==. Como su propio nombre indica, sirve para comparar dos valores o variables y no para asignar valores. Este operador devuelve el resultado de la operación lógica de 'igual a' tal y como podemos apreciar en el siguiente ejemplo.

### Ejemplo

```
<?php
    $a = 3;
    $b = 3;
    $a == $b; // El resultado de la comparación es TRUE porque $a es igual a $b
    $a = 3;
    $b = 4;
    $a == $b; // El resultado de la comparación es FALSE porque $a no es igual a $b
?>
```

## TIPOS DE DATOS

En PHP no es obligatorio indicar el tipo de datos al que pertenece una variable como en otros lenguajes, sino que los tipos de datos son establecidos directamente por el intérprete PHP, que es el encargado de interpretar el código.

Ahora veremos en una tabla los distintos tipos de datos que se pueden usar en PHP.

| TIPO DE DATO          | DEFINICIÓN  |
|-----------------------|---|
| <b>integer</b>        | <p>Los integers, o enteros, pueden tener distintos valores numéricos enteros que se expresan con diferentes notaciones.</p> <p><code>\$variable = 18;</code> // Número entero positivo</p> <p><code>\$variable = -18;</code> // Número entero negativo</p> <p><code>\$variable = 0x12;</code> // Notación hexadecimal, en principio no la utilizaremos.</p> |
| <b>float o double</b> | <p>Este tipo de datos son los números de punto flotante a los que normalmente llamamos "números decimales", por ejemplo, 9.876. Ambos tienen mucha precisión, pero double es el más preciso (con más decimales). La sintaxis para utilizarlos es bastante simple:</p> <p><code>\$variable = 9.876;</code></p>   |
| <b>string</b>         | <p>El tipo de datos string, también conocido como cadena de caracteres, se expresa con la siguiente sintaxis:</p> <p><code>\$variable = "Yo soy una cadena";</code></p>   |
| <b>boolean</b>        | <p>Se trata de un tipo lógico. Sus posibles valores son true (verdadero) o false (falso).</p> <p><code>\$variable = true;</code></p> <p><code>\$variable = false;</code></p>  |

## INSTRUCCIÓN ECHO

Esta instrucción muchas veces se dice que sirve para que se muestre por pantalla un texto, una variable, o una combinación de texto y variables. Pero realmente no es así: echo no es una instrucción que sirva para mostrar algo por pantalla (aunque podemos usarla para esto), sino que sirve para insertar texto dentro del documento HTML subyacente al código PHP.

Por ejemplo: `echo "salida por pantalla. ";` hará que se inserte este texto en el documento html. Si lo hacemos entre las etiquetas `<body>` y `</body>` lo veremos por pantalla porque cualquier texto insertado en ese lugar se mostrará por pantalla, no porque la instrucción echo dé lugar a que se muestre por pantalla.

Sin embargo `echo "<h1>Salida por pantalla. </h1> ";` no hará que se muestren por pantalla las etiquetas h1 y el texto, sino que introducirá en el documento HTML las etiquetas con el texto, éstas etiquetas funcionarán como código HTML y lo que se mostrará por pantalla será **Salida por pantalla** con formato de título h1 (letras más grandes de lo normal).

En general en php son válidas tanto las comillas dobles como las comillas simples, por lo que se puede escribir tanto `echo "<h1>Salida por pantalla. </h1> ";` como `echo '<h1>Salida por pantalla. </h1>';`

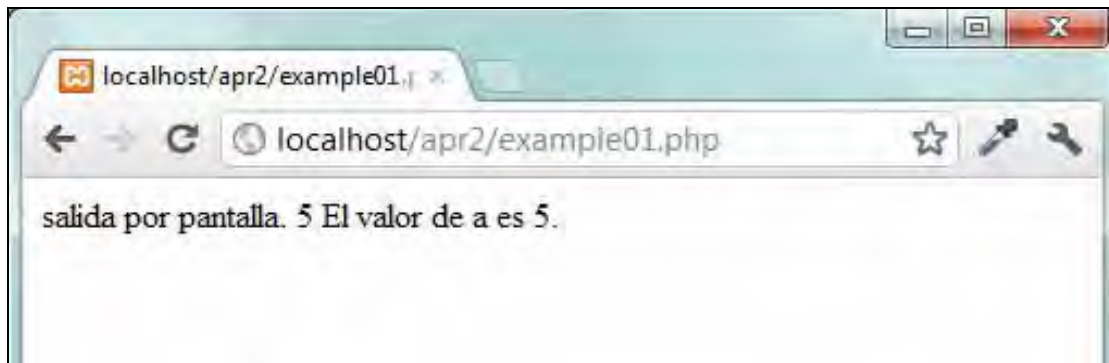
Los paréntesis no son necesarios (echo en realidad no es una función sino que es lo que se denomina una <<construcción del lenguaje>>, de ahí que no sean estrictamente necesarios los paréntesis), pero se pueden incluir si se desea. Ejemplo:

echo ('con paréntesis y comillas simples');

Crea un archivo php con el siguiente código, ponle un nombre como usoDeEcho.php y súbelo al servidor. Luego invócalo usando la ruta adecuada según la dirección web con la que estés trabajando (será similar a <http://apr2.byethost7.com/usoDeEcho.php>.)

```
<?php
    echo "salida por pantalla. ";
    $a = 5;
    echo $a;
    echo " El valor de a es $a.";
?>
```

La visualización en tu navegador deberá ser similar a esta. Puede variar ligeramente dependiendo del navegador que uses.



Ten en cuenta que \$a se muestra por pantalla como 5 porque el intérprete de PHP considera que debe mostrarse siempre el valor de la variable, incluso si va dentro de un texto. Prueba a escribir la siguiente línea: echo " El valor de la variable \ \$a es \$a.";

Comprobarás que se muestra por pantalla "salida por pantalla. 5 El valor de la variable \$a es 5."

Cuando antecede un nombre de variable del carácter \ hacemos que se interprete como texto en lugar de como una variable. El carácter \ se suele llamar "carácter de escape" porque permite que se considere texto algo que en principio se iba a interpretar como otra cosa (como una variable por ejemplo).

Crea ahora un archivo php con el siguiente código, ponle un nombre como usoDeEcho2.php y súbelo al servidor. Luego invócalo usando la ruta adecuada según la dirección web con la que estés trabajando (será similar a <http://apr2.byethost7.com/usoDeEcho2.php>.)

```
<html>
<head>
<title> Ejemplo php aprenderaprogramar.com </title>
<meta charset="utf-8">
</head>
<body>
<?php
echo 'con comillas simples<br/>';
echo ('con paréntesis y comillas simples<br/>');
echo ("con paréntesis y comillas dobles<br/>");
echo '<h1>Salida por pantalla</h1><br/>';
echo "salida por pantalla. ";
$a = 5;
echo $a;
echo " El valor de a es $a.";
?>
</body>
</html>
```

La visualización en tu navegador deberá ser similar a:

con comillas simples  
 con paréntesis y comillas simples  
 con paréntesis y comillas dobles  
**Salida por pantalla**  
 salida por pantalla. 5 El valor de a es 5.

Puedes comprobar que `<br/>` no se muestra por pantalla porque es código HTML, y el navegador cuando detecta este código HTML introduce un salto de línea en lugar de mostrarlo por pantalla.

Para comprender bien la utilidad de la instrucción echo piensa en esta equivalencia:

|   |                                   |  |
|---|-----------------------------------|--|
| <pre>&lt;body&gt; &lt;? php echo '&lt;h1&gt;Cuzco&lt;/h1&gt;' ?&gt; &lt;/body&gt;</pre> | <p>----- Equivale a -----&gt;</p> | <pre>&lt;body&gt; &lt;h1&gt;Cuzco&lt;/h1&gt; &lt;/body&gt;</pre> |
|---|-----------------------------------|--|

El navegador recibe el HTML "equivalente" generado y lo interpreta, dando lugar a los resultados que se mostrarán por pantalla.

## EJERCICIO 1

Crea el siguiente archivo y súbelo mediante ftp al directorio raíz de tu sitio web. El archivo llevará por nombre curso3.php y en él debe existir la estructura de un documento HTML, aparecer en la parte superior de la página un título h1 con el texto <<Cursos aprenderaprogramar.com>>, seguido de un título h2 con el texto <<Curso php desde cero>>, y a continuación hacer que se muestre usando echo el siguiente texto dentro de un div con color de fondo amarillo: <<Curso php de aprenderaprogramar.com usando la instrucción echo>>. Debajo de este texto debes usar echo para crear un contenedor div con color de gris donde pueda leerse "Este es el pie de página". Comprueba que el archivo se visualiza al escribir la dirección web correcta que será similar a <http://apr2.byethost7.com/curso3.php> (en tu caso, tendrás que escribir la dirección web que estés utilizando para el curso). Responde a esta pregunta: ¿Cuál es el código que contiene el archivo? ¿Cuál es la dirección web para acceder al archivo y visualizar el texto?

Para comprobar si tus respuestas y código son correctos puedes consultar en los foros [aprenderaprogramar.com](http://aprenderaprogramar.com).

## EJERCICIO 2

Crea un archivo php donde no exista html directamente pero que genere una página web con todo el código html que lleva normalmente (introduce las etiquetas html, head, title, meta charset, body, algunos párrafos de texto y algunas imágenes). Para hacer esto toda la estructura de la página web debe ser insertada a través de instrucciones echo. Responde a esta pregunta: ¿Cuál es el código que contiene el archivo? ¿Cuál es la dirección web para acceder al archivo y visualizar el texto?

Para comprobar si tus respuestas y código son correctos puedes consultar en los foros [aprenderaprogramar.com](http://aprenderaprogramar.com).

**Próxima entrega:** CU00817B

**Acceso al curso completo** en [aprenderaprogramar.com](http://aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente:  
[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=70&Itemid=193](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=70&Itemid=193)