

PHP - Operadores

Un operador es “algo” que toma uno valores o expresiones, en la jerga de programación y produce otro valor (de modo que la construcción en sí misma se convierte en una expresión).

Los operadores se pueden agrupar de acuerdo con el número de valores que toman:

- Los operadores unarios toman sólo un valor, por ejemplo `!` (el **operador lógico de negación**)
o `++` (el **operador de incremento**).
- Los operadores binarios toman dos valores, como los familiares **operadores aritméticos** `+` (suma) y `-` (resta), y la mayoría de los operadores de PHP entran en esta categoría.
- Finalmente, hay sólo un **operador ternario**, `?:`, el cual toma tres valores;
 - usualmente a éste se le refiere simplemente como "el operador ternario" (aunque podría tal vez llamarse más correctamente como el operador condicional).

Ejemplo:

```
$mensaje = ($dinero > 10) ? 'Me puedo ir al cine!' : 'Me tengo que
quedar en casa... sniff';
```

Operador de asignación

Al igual que en otros lenguajes el operador de asignación que emplea PHP es el símbolo igual (=)

```
<?php
    $variable1 = 1234;
    $variable2 = "abcdef";
?>
```

Operadores de asignación combinada

Permiten realizar en una misma acción una operación y una asignación

Ejemplo	Nombre	Resultado
<code>\$a++</code>	Incremento	<code>\$a = \$a + 1</code>
<code>\$a--</code>	Decremento	<code>\$a = \$a - 1</code>
<code>++\$a</code>	Incremento	<code>\$a = \$a + 1</code>
<code>--\$a</code>	Decremento	<code>\$a = \$a - 1</code>
<code>\$a += \$b</code>	Suma	<code>\$a = \$a + \$b</code>
<code>\$a -= \$b</code>	Resta	<code>\$a = \$a - \$b</code>
<code>\$a *= \$b</code>	Multiplicación	<code>\$a = \$a * \$b</code>
<code>\$a /= \$b</code>	División	<code>\$a = \$a / \$b</code>
<code>\$a %= \$b</code>	Módulo	<code>\$a = \$a % \$b</code>
<code>\$a &= \$b</code>	Y	<code>\$a = \$a & \$b</code>
<code>\$a = \$b</code>	O	<code>\$a = \$a \$b</code>
<code>\$a ^= \$b</code>	O exclusiva	<code>\$a = \$a ^ \$b</code>
<code>\$a .= \$b</code>	Concatenación	<code>\$a = \$a . \$b</code>
<code>\$a >>= \$b</code>	Desplazamiento a la derecha	<code>\$a = \$a >> \$b</code>
<code>\$a <<= \$b</code>	Desplazamiento a la izquierda	<code>\$a = \$a << \$b</code>

Veamos ejemplos de los operadores de asignación “++” (incremento) y “--” (decremento).

Ejemplo:

```
$x++;      //esto es $x=$x+1;  
$x--;      //esto es $x=$x-1;
```

Pero hay dos formas de utilizar el incremento y el decremento. Se puede usar por ejemplo **x++** o **++x**

La diferencia radica en el modo en el que se comporta la asignación.

Ejemplo:

```
$x=5;  
$y=5;  
$z=$x++;      //z vale 5, x vale 6  
$z=++$y;      //z vale 6, y vale 6
```

Operador unario

Aunque no seamos conscientes de ello el operador menos (-), opera sobre los números o variables cambiándoles el signo

```
<?php  
    $variable1 = 1234;  
    $variable2 = -$variable1; // variable2 ahora vale -1234  
    $variable3 = -$variable2; // variable3 ahora vale 1234  
?>
```

PHP - Operadores aritméticos

Estos se pueden aplicar a las variables y a los números. Son los operadores que vulgarmente conocemos como matemáticos.

Operador	Nombre	Operación
\$a + \$b	suma	Suma las dos variables.
\$a - \$b	resta	Diferencia de las dos variables.
\$a * \$b	multiplicación	Producto de las variables.
\$a / \$b	división	Cociente entre las dos variables.
\$a % \$b	módulo	Resto de la división de \$a entre \$b.

Visto de otro modo:

Operador	Nombre	Ejemplo	Descripción
+	Suma	5 + 6	Suma dos números
-	Resta	7 - 9	Resta dos números
*	Multiplicación	6 * 3	Multiplica dos números
/	División	4 / 8	Divide dos números
%	Módulo	7 % 2	Devuelve el resto de dividir ambos números, en este ejemplo el resultado es 1
++	Suma 1	\$a++	Suma 1 al contenido de una variable.
--	Resta 1	\$a--	Resta 1 al contenido de una variable.

PHP - Operadores de comparación o condicionales

Estos operadores sirven para comparar valores. Siempre devuelven valores booleanos. Son los siguientes:

Operador	Nombre	Operación
\$a == \$b	Igualdad	Verdadero si \$a igual que \$b
\$a === \$b	Identidad	Verdadero si \$a idéntico que \$b
\$a <> \$b	NO iguales	Verdadero si \$a distinto de \$b
\$a != \$b	NO iguales	Verdadero si \$a distinto de \$b
\$a > \$b	Mayor que	Verdadero si \$a mayor que \$b
\$a < \$b	Menor que	Verdadero si \$a menor que \$b
\$a >= \$b	Mayor o igual que	Verdadero si \$a mayor o igual que \$b
\$a <= \$b	Menor o igual que	Verdadero si \$a menor o igual que \$b

Ejemplo:

```
$a = 8;
$b = 3;
$c = 3;
echo ($a == $b) && ($c > $b), "<br>";
echo ($a == $b) || ($b == $c), "<br>";
echo !($b <= $c), "<br>";
```

Operadores Lógicos

Los operadores lógicos son aquellos que nos permiten concatenar de una forma determinada ciertas comparaciones, esto es: evalúan condiciones complejas.

Operador	Nombre	Operación
\$a AND \$b	Y	Verdadero si ambos son verdaderos
\$a && \$b	Y	Verdadero si ambos son verdaderos
\$a OR \$b	O	Verdadero si alguno de los dos es verdadero
\$a \$b	O	Verdadero si alguno de los dos es verdadero
\$a XOR \$b	O exclusivo	Verdadero si sólo uno de los dos es verdadero
!\$a	Negación	Verdadero si \$a es falso

Ejemplo:

```
$edad = 21;  
$mayorDeEdad = $edad >= 18; //mayorDeEdad será true  
$menorDeEdad = !$mayorDeEdad; //menorDeEdad será false  
echo $mayorDeEdad."\t".$menorDeEdad;
```

El operador **&&** (AND) sirve para evaluar dos expresiones de modo que si ambas son ciertas, el resultado será **true** sino el resultado será **false**.

Ejemplo:

```
$carnetConducir=TRUE;  
$edad=20;  
$puedeConducir= ($edad>=18) && $carnetConducir;  
//puedeConducir es TRUE, puesto que edades mayor de 18  
//y carnet es TRUE
```

El operador **||** (OR) sirve también para evaluar dos expresiones. El resultado será **true** si al menos uno de las expresiones es **true**.

Ejemplo:

```
$nieva =TRUE;  
$llueve=FALSE;  
$graniza=FALSE;  
$malTiempo= $nieva || $llueve || $graniza; //malTiempo es TRUE porque nieva
```

La diferencia entre la igualdad y la equivalencia se puede explicar con este ejemplo.

Ejemplo:

```
$resultado= (18 == 18.0); //resultado es verdadero
```

```
$resultado= (18 === 18.0); //resultado es falso
```

En el ejemplo **18===18.0** devuelve falso porque aunque es el mismo valor, no es del mismo tipo.

Operador Ternario

Los operadores que hemos visto hasta ahora son capaces de manejar un operando (unarios) o dos operandos (binarios). El operador ternario, o de comparación, evalúa un operando y, dependiendo de si es falso o verdadero, evalúa el segundo operando o el tercero.

La expresión que se quiere evaluar se escribe delante de un símbolo (?), después la expresión que tiene que ejecutarse si la evaluación anterior es true, seguida del símbolo (:) con la expresión que debe ejecutarse si es false.

```
<?php
    $valor = false;
    $valor == true ? $resultado = "OK" : $resultado = "FALLO";
    // Si $value es true $resultado será OK
    // Si es false $resultado será FALLO
    echo $resultado;
?>
```

Operador de concatenación

El punto (.) es un operador que permite unir textos. Su uso es muy sencillo.

Ejemplo:

```
$x="Hola ";
```

```
$y="a todo el mundo";
```

```
$z=$x.$y; //$z vale "Hola a todo el mundo"
```

Hay operadores de asignación y concatenación de texto, se trata del operador .=.

Ejemplo:

```
$x="Hola ";
```

```
$x.="a todo el mundo";
```

```
echo $x; //escribe "Hola a todo el mundo"
```

También es válido el uso de la concatenación en la asignación.

Ejemplo:

```
$texto = "Hola ";
```

```
$texto.= "amigos y ";
```

```
$texto.="amigas";           //$texto es Hola amigos y amigas
```

Precedencia de operadores:

Operador	Operación
()	Paréntesis de preferencia
New	Instancia de objeto
[]	Array
!	NO lógico
-	Signo menos
++ --	Incremento, decremento
@	Supresión de errores
· / %	Multiplicación, división y módulo
+ - .	Suma, resta, concatenación
<< >>	Desplazamiento izquierda y derecha
< <=	Menor que, menor o igual
> >=	Mayor que, mayor o igual
== !=	Igual, no igual
&	Y
^	O exclusivo
	O
&&	Y lógico
	O lógico
?:	Condicional
= += -= *= /=	Asignación
= %= &= !=	Asignación
= <<= >>=	Asignación
And	Y Lógico
XOR	O Exclusivo
OR	O Lógico