

Está muy bien definir tipos de datos, pero debemos poder también realizar operaciones con ellos. Para ello contamos con diferentes operadores. Por la cantidad de datos que necesitan, los podemos clasificar en:

- Operadores unarios: - (signo menos delante de un número), ! (not), ++, –
- Operadores binarios: +, -, *, /, ^, ., |, &, ||, &&, ==, =, !=, !, <, >*
- Operadores trinaros: ? (comparación con asignación)

Por el tipo de dato sobre el que operan, los podríamos clasificar de otra manera:

- Operadores aritméticos, sobre enteros y flotantes
 - +, -, *, /, ^
- Operadores binarios, para números binarios
 - | (OR), & (AND), ! (NOT)
- Operadores binarios en “cortocircuito” y pruebas lógicas
 - || (OR), && (AND), ==, =, !=, !, <, >
- Operadores para cadenas
 - . (concatenar)

Los operadores tienen cierta **precedencia** que puede ser alterada utilizando paréntesis.

Al realizar una operación sobre una variable, y almacenar el resultado sobre la misma variable, tenemos una forma corta para expresar estas asignaciones:

```
$a = 5;
//Le sumamos 3
$a = $a + 3;
//Lo anterior es equivalente a esto (aunque lo estamos repitiendo dos veces)
$a += 3;

$m = 'Hola';
$m = $m. ' mundo';
//Lo anterior es equivalente a esto (aunque lo estamos repitiendo dos veces)
$m .= ' mundo';
```

Una asignación es una operación a su vez, que devuelve un valor que puede ser utilizado para otra asignación.

```
$c = 3;
$a = $b = $c + 1;
//Ahora tanto $a como $b valen 4
```

Los operadores unarios ++ y – realizan una asignación y suma o resta de 1 en una variable, por lo que las siguientes expresiones son equivalentes

```
$a = 0; //Para empezar con algún valor
$a = $a + 1;
$a += 1;
$a++;

$b = 0;
$b = $b - 1;
```

```
$b -= 1;  
$b--;
```

Si el operador ++ o – se antepone a la variable, primero se realiza la operación de sumar o restar uno, y luego se devuelve el resultado. Si se pospone (como en los ejemplos anteriores), primero se devuelve el valor original de la variable, y después se realiza el incremento o decremento.

```
$a = 1;  
$b = $a++;  
echo $b; //1  
  
$a = 1;  
$c = ++$a;  
echo $c //2
```