

## Operadores de asignación

Los operadores de asignación almacenan un valor en el objeto especificado por el operando izquierdo. Hay dos tipos de operaciones de asignación:

- *asignación simple*, en la que el valor del segundo operando se almacena en el objeto especificado por el primer operando.
- *asignación compuesta*, en la que se realiza una operación aritmética, de desplazamiento o bit a bit antes de almacenar el resultado.

Todos los operadores de asignación de la tabla siguiente, salvo el operador `=`, son operadores de asignación compuesta.

### Tabla de operadores de asignación

Operador	Significado
<code>=</code>	Almacena el valor del segundo operando en el objeto especificado por el primer operando (asignación simple).
<code>*=</code>	Multiplica el valor del primer operando por el valor del segundo operando; almacena el resultado en el objeto especificado por el primer operando.
<code>/=</code>	Divide el valor del primer operando por el valor del segundo operando; almacena el resultado en el objeto especificado por el primer operando.
<code>%=</code>	Toma el módulo del primer operando especificado por el valor del segundo operando; almacena el resultado en el objeto especificado por el primer operando.
<code>+=</code>	Suma el valor del segundo operando al valor del primer operando; almacena el resultado en el objeto especificado por el primer operando.
<code>-=</code>	Resta el valor del segundo operando del valor del primer operando; almacena el resultado en el objeto especificado por el primer operando.
<code>&lt;&lt;=</code>	Desplaza a la izquierda el valor del primer operando el número de bits especificado por el valor del segundo operando; almacena el resultado en el objeto especificado por el primer operando.
<code>&gt;&gt;=</code>	Desplaza a la derecha el valor del primer operando el número de bits especificado por el valor del segundo operando; almacena el resultado en el objeto especificado por el primer operando.
<code>&amp;=</code>	Obtiene el AND bit a bit del primer y el segundo operandos; almacena el resultado en el objeto especificado por el primer operando.
<code>^=</code>	Obtiene el OR exclusivo bit a bit del primer y el segundo operandos; almacena el resultado en el objeto especificado por el primer operando.
<code> =</code>	Obtiene el OR inclusivo bit a bit del primer y el segundo operandos; almacena el resultado en el objeto especificado por el primer operando.

### Ejemplo

```
// expre_Assignment_Operators.cpp
// compile with: /EHsc
// Demonstrate assignment operators
#include <iostream>
using namespace std;
int main() {
    int a = 3, b = 6, c = 10, d = 0xAAAA, e = 0x5555;

    a += b;      // a is 9
    b %= a;      // b is 6
    c >>= 1;     // c is 5
```

```
d |= e;      // Bitwise--d is 0xFFFF

cout << "a = 3, b = 6, c = 10, d = 0xAAAA, e = 0x5555" << endl
    << "a += b yields " << a << endl
    << "b %= a yields " << b << endl
    << "c >>= 1 yields " << c << endl
    << "d |= e yields " << hex << d << endl;
}
```

## Bibliografía:

- Extraído de ["<<Operadores Asignación>>"](#). 26 de septiembre de 2022. Consultado el 28 de noviembre del 2022.