|  |  |  |  |
| --- | --- | --- | --- |
| **a** | **b** | **a && b** | **a || b** |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 |

int x, y;

// x e y tienen un valor no conocido

// usar el operador ternario para devolver el valor mayor de los dos

// condición ? valor1 : valor2

x>y ? x : y;

/\* Dadas dos variables String usar el operador ternario para devolver el valor de una de ellas sin son iguales y null si son diferentes (método equals)\*/

String var1, var2;

// var1 y var2 tienen un valor no conocido

var1.equals(var2) ? var1 : null;

//Dado dos números de tipo int sume ambos valores si el primer número es menor que el //segundo número y reste el segundo número del primero en caso contrario

int resultado;

if (x<y) {

resultado = x+y;

}

else {

resultado=y-x;

}

/\* sume ambos valores si el primer número es menor que el segundo número, reste el segundo número del primero si el primer número es mayor y almacene en el resultado uno de los dos números si son iguales\*/

int resultado;

if (x<y) {

resultado = x+y;

}

else if (x>y) {

resultado=y-x;

} else {

resultado= x;

}

// ejercicio transparencia 33 (switch)🡪 pendiente (opcional)

// ejercicio transparencia 39 (for)🡪 pendiente (hacer al menos un enunciado)

// ejercicio transparencia 45 🡪 pendiente (opcional). Operadores de división (/) y de módulo (%)

int a= 123;

int c= a % 10; // c guarda el valor de 3

int b= a/10; // b guarda el valor de 12

a=a/10; // a guarda el valor de 12

c= a % 10; // c guarda el valor de 2

a=a/10; // a guarda el valor de 1

c= a % 10; // c guarda el valor de 1