# ¿Existen nombres que no conviene utilizar al programar?

Al trabajar en nuestros programas hay una serie de palabras claves que tienen un significado importante en la programación, forman parte de estructuras que utilizamos habitualmente con diferentes finalidades, por lo que no se deben utilizar fuera de su ámbito de aplicación. A continuación se deja un listado de estas palabras.

|  |  |  |
| --- | --- | --- |
| And | finally | nonlocal |
| As | for | not |
| assert | from | or |
| break | False | pass |
| class | global | raise |
| continue | if | return |
| def | import | True |
| del | in | try |
| elif | is | while |
| else | lambda | with |
| except | None | yield |

# Operadores

Independientemente del lenguaje de programación utilizados, es necesario contar con operadores matemáticos, los más sencillos y que todos conocemos desde el colegio suelen ser el de suma (+), resta(-), multiplicación(\*) y división (/), pero existen muchos otros que pasaremos a describir a continuación separándolos en grupos.

Para los ejemplos que se presentan a continuación utilizaremos los siguientes valores de variables:

|  |
| --- |
| a, b, c, d, e, f = 2, 4, 5, 21, 22, 3 |

Notar como los valores pueden ser asignados considerando la posición que ocupan del lado izquierdo de la igualdad, con la correspondiente posición del lado derecho de la igualdad.

## Grupo 1 – Operadores Aritméticos

|  |  |  |  |
| --- | --- | --- | --- |
| **Operador** | **Descripción** | **Ejemplo** | **Resultado** |
| + | Suma | 5 + 7 | 12 |
| - | Resta | 7 - 5 | 2 |
| \* | Multiplicación | 2 \* 3 | 6 |
| / | División | 21 / 3 | 7 |
| % | Resto | 22 % 3 | 1 |
| // | División que retorna solo la parte entera | 22 // 3 | 7 |
| \*\* | Potencia | 4 \*\* 2 | 16 |

## Grupo 2 – Operadores de Comparación

|  |  |  |  |
| --- | --- | --- | --- |
| **Operador** | **Descripción** | **Ejemplo** | **Resultado** |
| > | Mayor que | print('b > a is',b>a) | b > a is True |
| < | Menor que | print('b < a is',b<a) | b < a is False |
| == | Igual a | print('b == a is',b==a) | b == a is False |
| != | Distinto de | print('b != a is',b!=a) | b != a is True |
| >= | Mayor o igual que | print('b >= a is',b>=a) | b >= a is True |
| <= | Menor o igual que | print('b <= a is',b<=a) | b <= a is False |