

PRINCIPALES SENTENCIAS DE CONTROL

Al igual que otros lenguajes de programación, Python incorpora una serie de sentencias de control. Entre ellas, encontramos algunas tan básicas y comunes a otros lenguajes como *if/else*, *while* y *for*, y otras específicas como *pass* y *with*. A continuación, echaremos un vistazo a cada una de estas sentencias.

if, else y elif

La sentencia *if/else* funciona evaluando la condición indicada, si el resultado es *True* se ejecutará la siguiente sentencia o sentencias, en caso negativo se ejecutarán las sentencias que aparecen a continuación del *else*. Recordemos que Python utiliza la indentación para establecer sentencias que pertenecen al mismo bloque. Además, en el carácter dos puntos (:) indica el comienzo de bloque. A continuación, vemos un ejemplo:

```
x = 4
y = 0
if x == 4:
    y = 5
else:
    y = 2
```

Obviamente, también es posible utilizar solo la sentencia *if* para comprobar si se cumple una determinada condición y actuar en consecuencia. Además, podemos anidar diferentes niveles de comprobación a través de *elif*:

```
x = 4
y = 0
if x == 4:
    y = 5
else:
    y = 2
```

Obviamente, también es posible utilizar solo la sentencia *if* para comprobar si se cumple una determinada condición y actuar en consecuencia. Además, podemos anidar diferentes niveles de comprobación a través de *elif*:

```
if x == 4:
    y = 1
elif x == 5:
    y = 2
elif x == 6:
    y = 3
else:
    y = 5
```

Como el lector habrá podido observar y a diferencia de otros lenguajes de programación, los paréntesis para indicar las condiciones han sido omitidos. Para Python son opcionales y habitualmente no suelen ser utilizados. Por otro lado, a pesar de que Python emplea la indentación, también es posible escribir una única sentencia a continuación del final de la condición. Así pues, la siguiente línea de código es válida:

```
if a > b: print("a es mayor que b")
```