

NIVEL 2

BOOLEANOS & SUS OPERADORES
(EXPRESIONES RELACIONALES Y LÓGICAS)

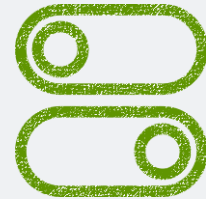




RECORDAMOS...

TIPO BOOLEANO O LÓGICO

- Un dato de tipo lógico solo puede presentar uno de dos valores: **True** o **False**, es decir, verdadero o falso
- Como con los demás tipos de datos que hemos visto (**int**, **float** o **string**), pueden existir expresiones, variables y funciones con valores booleanos (**bool**)



EXPRESIONES RELACIONALES

Las expresiones relacionales se construyen con operadores relacionales (de orden, igualdad y diferencia)

Operando

- Valores
- Variables
- Parámetros
- Llamadas a funciones
- Otras expresiones

Operador relacional

Operando

- Valores
- Variables
- Parámetros
- Llamadas a funciones
- Otras expresiones

OPERADORES RELACIONALES



Es igual que	<code>==</code>	<code>x == y</code> es True si x es igual a y
Es diferente de	<code>!=</code>	<code>x != y</code> es True si x es diferente de y
Es menor que	<code><</code>	<code>x < y</code> es True si x es menor que y
Es mayor que	<code>></code>	<code>x > y</code> es True si x es mayor que y
Es menor o igual que	<code><=</code>	<code>x <= y</code> es True si x es menor o igual que y
Es mayor o igual que	<code>>=</code>	<code>x >= y</code> es True si x es mayor o igual que y

EJEMPLOS

Note la diferencia entre la asignación `=` y a la expresión con el operador `>=`



```
In [43]: 5 == (3+2)
```

```
Out[43]: True
```

```
In [44]: 5 == 6
```

```
Out[44]: False
```

```
In [45]: "hola" > "hoja"
```

```
Out[45]: True
```

```
In [46]: "Z" < "a"
```

```
Out[46]: True
```

```
In [47]: edad = 18
```

```
In [48]: edad_suficiente_para_sacar_pase = edad >= 16
```

```
In [49]: type(edad_suficiente_para_sacar_pase)
```

```
Out[49]: bool
```

```
In [50]: 0.2 == 1.2 - 1.0
```

```
Out[50]: False
```

```
In [51]: round(0.2, 1) == round(1.2-1.0, 1)
```

```
Out[51]: True
```

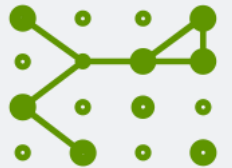
OPERADORES DE IDENTIDAD

- Los operadores **is** y **is not** sirven para revisar que dos valores no sólo sean iguales, sino que también sean el mismo.
- En este curso no nos ocuparnos mucho de objetos, no ahondaremos en la distinción entre **is** y **==**



EXPRESIONES LÓGICAS

- Permiten describir situaciones más complejas que las expresiones relacionales
- Son composiciones de expresiones relacionales o de valores, variables, parámetros o funciones de tipo boolean



OPERADORES LÓGICOS



Operador	Se lee como
and (conjunción)	y
or (disyunción)	o
not (negación)	no

La semántica (significado) de estos operadores es similar a su significado en español. Por ejemplo: $x > 0$ and $x < 10$ es verdadero (**True**) solo si x es mayor que 0 y al mismo tiempo, x es menor que 10

operando1 and operando2	Es cierto, si ambos operandos son verdaderos
operando1 or operando2	Es cierto, si cualquiera de los dos operandos es verdadero
not operando1	Es cierto, si el operando es falso

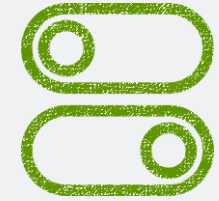
EJEMPLO

Tenemos tres proposiciones P, Q y R:

- P: Hoy está haciendo calor
- Q: Hoy es martes
- R: Hoy es un día festivo

- Q: Hoy es martes.
- not P: Hoy no está haciendo calor
- not Q: Hoy no es martes
- P and Q: Hoy está haciendo calor y es martes
- Q and not R: Hoy es martes y no es un día de fiesta
- P or R: Hoy está haciendo calor o es un día de fiesta

OTRO EJEMPLO



Para saber si algo es verdadero o falso

La expresión ...	Da como valor ...
$n \% 2 == 0 \text{ or } n \% 3 == 0$	es True si una de las dos condiciones es True , esto es, si el número n es divisible por 2 o es divisible por 3
$\text{not } (x > y)$	es True si $(x > y)$ es False , esto es, si x es menor a igual a y