

# Lógica

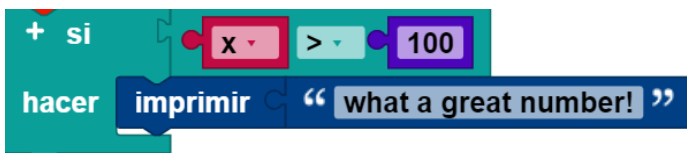
## Instrucciones condicionales

Las instrucciones son fundamentales para realizar la programación. Permiten realizar distinciones entre casos para la formulación, como:

- Si hay un camino a la izquierda, gire a la izquierda.
- Si la puntuación = 100, pulse «¡Bien hecho!»

### Bloques si

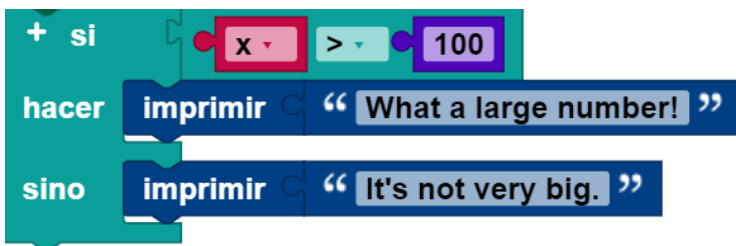
El **bloque si** incluye una condición sencilla:



Si este se implementa, el valor de la variable **x** se compara con 100. Cuando el valor es superior aparece «¡Qué cifra más alta!» En caso contrario, no se muestra nada.

### Bloques si-en caso contrario

También es posible indicar lo que se desea que suceda, en caso de que no se cumpla la condición, como en este ejemplo:

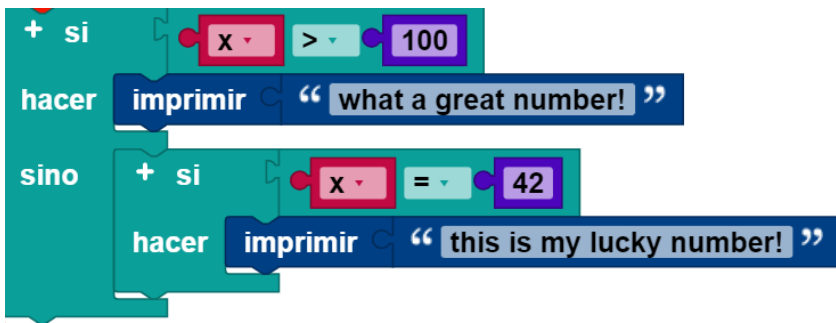


Como en el bloque anterior, se muestra «¡Qué cifra más alta!» cuando  $x > 100$ . De lo contrario, aparecerá «La cifra no es muy alta».

Un **bloque si** tiene un **apartado** en caso contrario, pero no más de uno.

### Bloques si-en caso contrario-si

También es posible comprobar más condiciones con un único bloque **si**, en el que se pueden añadir cláusulas **en caso contrario-si**:

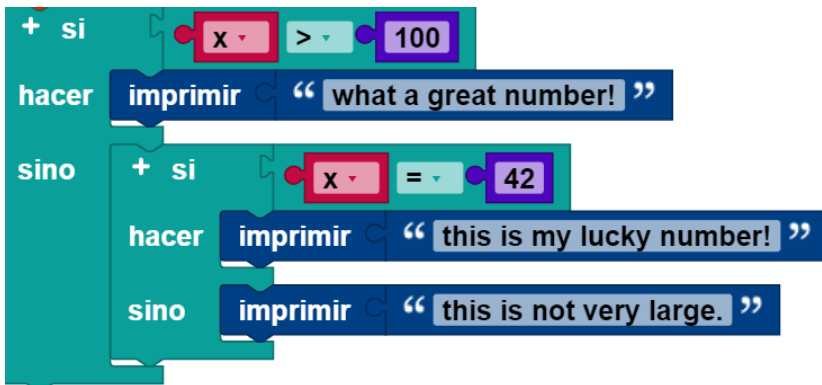


El bloque examina primero si  $x > 100$ , y muestra «¡Qué cifra más alta!» en caso de que corresponda. Si no se da el caso, vuelve a comprobar si  $x = 42$ . Si ese es el caso, aparece «¡Ese es mi número de la suerte!» En caso contrario, no se muestra nada.

Un bloque **si** puede tener un número discrecional de apartados **en caso contrario-si**. Las condiciones se evalúan en orden descendente hasta que se cumple una de ellas o hasta que no quedan más condiciones.

## Bloques **si-en caso contrario-si-en caso contrario**

Los bloques **si** pueden tener apartados **en caso contrario-si** y apartados **en caso contrario**:

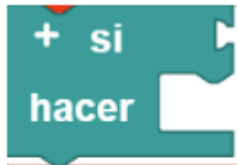


El apartado **en caso contrario** garantiza que se lleve a cabo una acción aunque no se cumplan las condiciones anteriores.

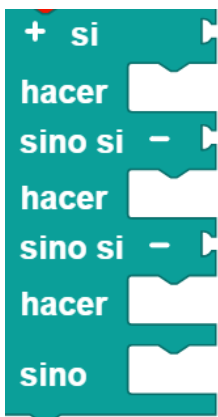
El apartado **en caso contrario** puede aparecer después de un número discrecional de secciones **en caso contrario-si**, incluido el cero. Después se obtiene un bloque completamente normal **si-en caso contrario**.

## Modificación de bloques

En la barra de herramientas solamente aparece el bloque sencillo **si** y el bloque **si-en caso contrario**:



Para añadir las cláusulas **en caso contrario-si** - y **en caso contrario**, haga clic en el símbolo (+). Con el símbolo (-) puede quitar las cláusulas **en caso contrario-si**:



Tenga en cuenta que las formas de los bloques permiten añadir cualquier número de subbloques **en caso contrario-si**, pero únicamente un bloque **si**.

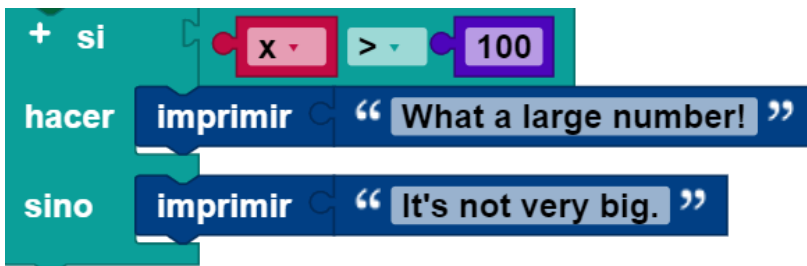
## Lógica booleana

La lógica booleana es un sistema matemático simple que tiene dos valores:

- verdadero
- falso

Los bloques lógicos en ROBO Pro Coding se utilizan generalmente para controlar condiciones y bucles.

El siguiente es un ejemplo:



Cuando el valor de la variable *x* es superior a 100, cuando se cumple la condición y se muestra el texto «¡Qué cifra más alta!» Si el valor de *x* no es superior a 100, no se cumple la condición y se muestra «La cifra no es muy alta». Los valores booleanos también se pueden almacenar en variables y pasar a funciones, al igual que los números, el texto y los valores de lista.

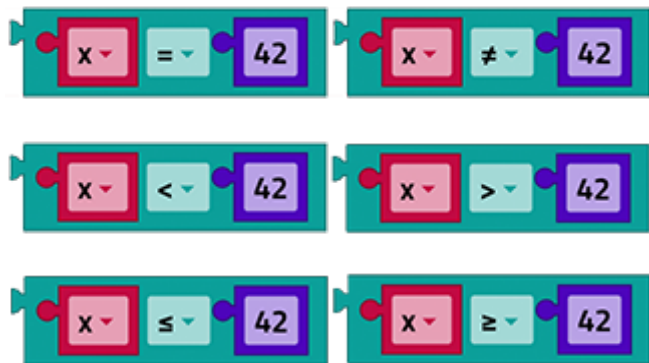
Si en un bloque se ha de introducir un valor booleano, cuando falta la entrada, esta se interpreta como **falsa**. Los valores no booleanos no se pueden insertar directamente en el lugar de los valores booleanos, aunque es posible (pero no recomendable) almacenar un valor no booleano en una variable y después insertarlo en la entrada que hace referencia a una condición concreta. Este método no se recomienda y su comportamiento puede cambiar en futuras versiones de ROBO Pro Coding.

## Valores

Se puede utilizar un único bloque con una lista desplegable que muestre **verdadero** o **falso** para acceder a un valor booleano:

## Operadores comparativos

Hay seis operadores comparativos. A cada uno se le otorgan dos entradas (normalmente dos cifras) y el operador comparativo muestra **verdadero** o **falso**, dependiendo de cómo se comparen las entradas.



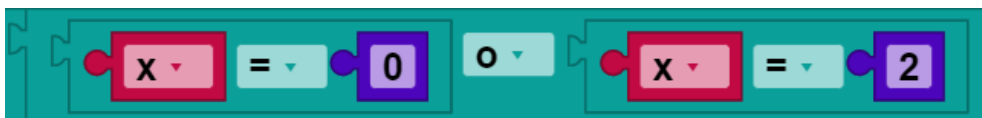
Los seis operadores son: igual, diferente, inferior a, superior a, inferior a o igual y superior a o igual.

## Operadores lógicos

El bloque **y** muestra **verdadero** únicamente cuando se cumplen ambos valores de entrada.



El bloque **o** muestra **verdadero** cuando se cumple, como mínimo, uno de sus dos valores de entrada.



## no

El bloque **no** convierte una entrada booleana en su opuesto. Por ejemplo, esto es resultado de:



falso.

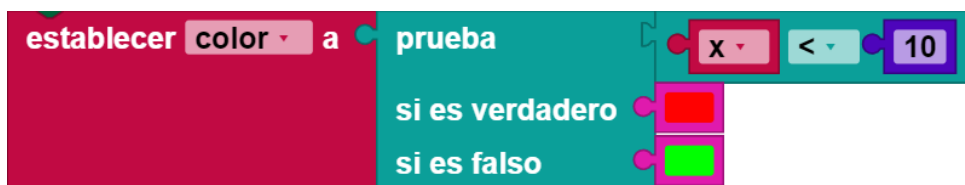
Si no hay ninguna entrada, se asume que el valor es **verdadero** por lo que el siguiente bloque genera el valor **falso**:



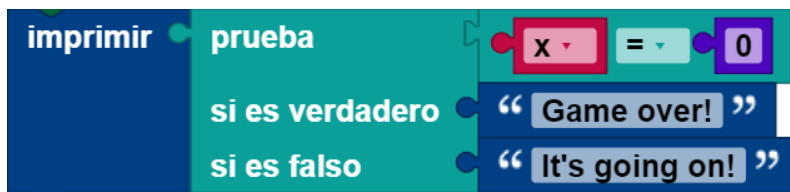
Sin embargo, no se recomienda dejar una entrada vacía.

## Operador tres

El operador tres se comporta como un bloque en miniatura **si-en caso contrario**. Se necesitan tres valores de entrada: el primer valor de entrada es la condición booleana que se va a probar, el segundo valor de entrada es el valor que debe mostrarse si la prueba da como resultado **verdadero**, y el tercer valor de entrada es el valor que debe aparecer si la prueba muestra falso. En el siguiente ejemplo, la variable **color** se configura en rojo, si la variable **x** es inferior a 10. En caso contrario, la variable **color** se configura en verde.



Un bloque tres siempre se puede reemplazar por un bloque **si-en caso contrario**. Los dos ejemplos siguientes son exactamente iguales.



Revision #11

Created 17 November 2021 20:25:15 by Admin

Updated 10 November 2024 15:09:08 by phuesing