

# Luces en micro:bit con Makecode

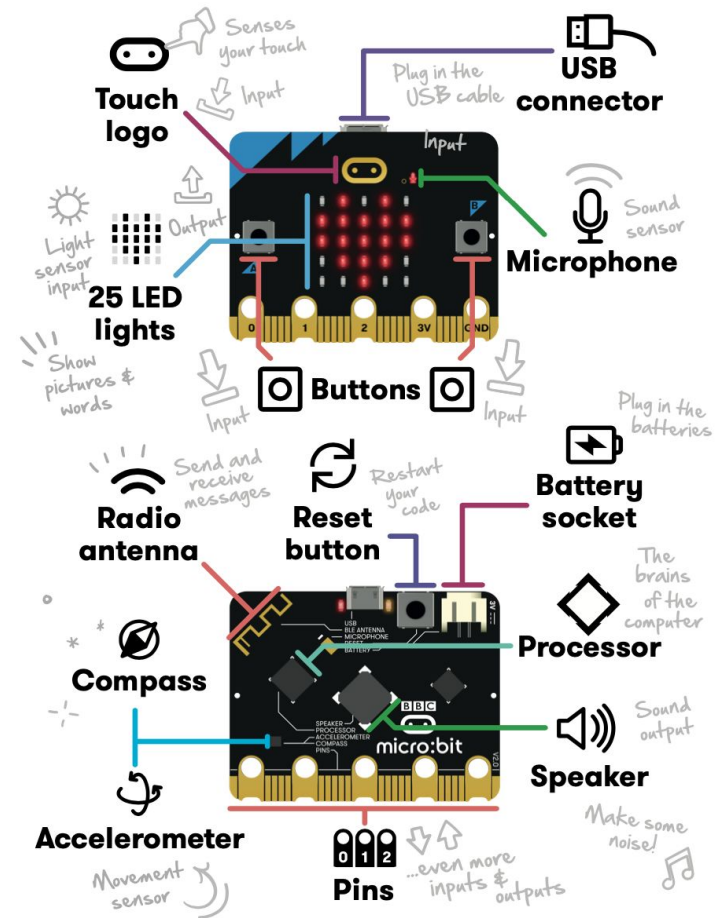
ZER Moianès Llevant 2023

# PRESENTACIÓN

La placa **micro:bit** permite crear letras, números y dibujos de una forma sencilla.

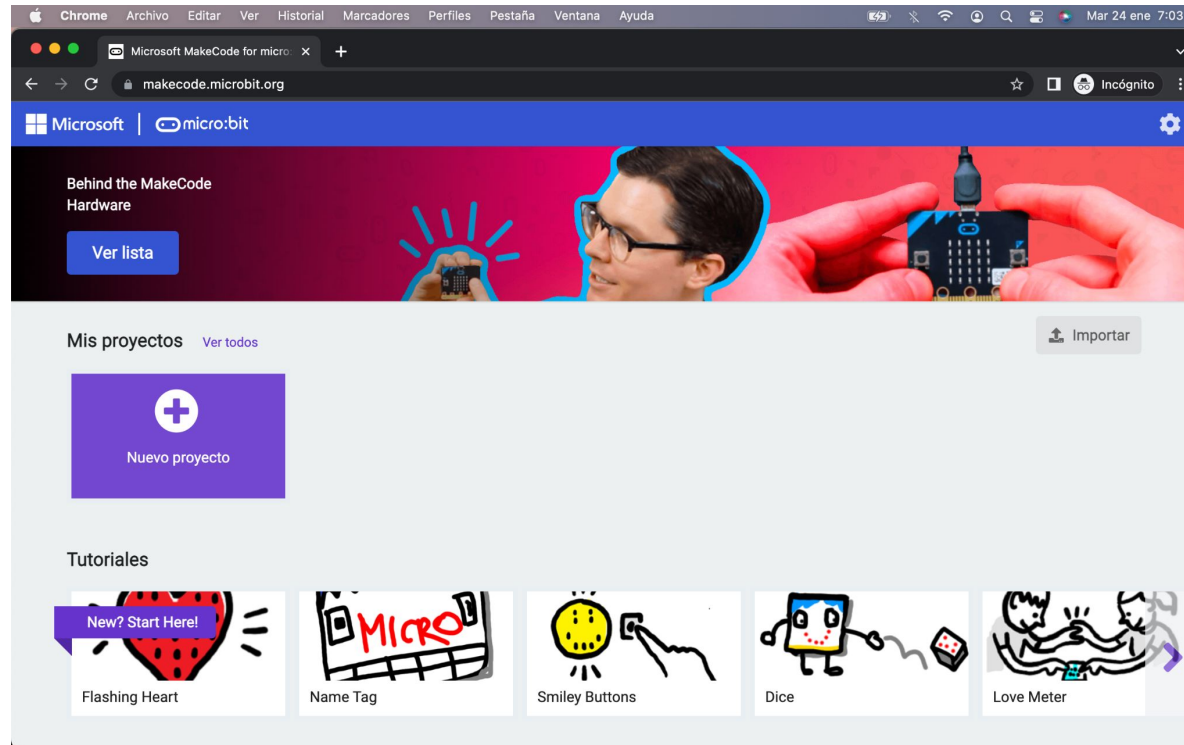
Trataremos de hacer una luz que se encienda sola al hacerse oscuro por la noche

Veamos como hacerlo en el simulador de Makecode y cómo subirlo al **micro:bit**



# Empezando con MakeCode

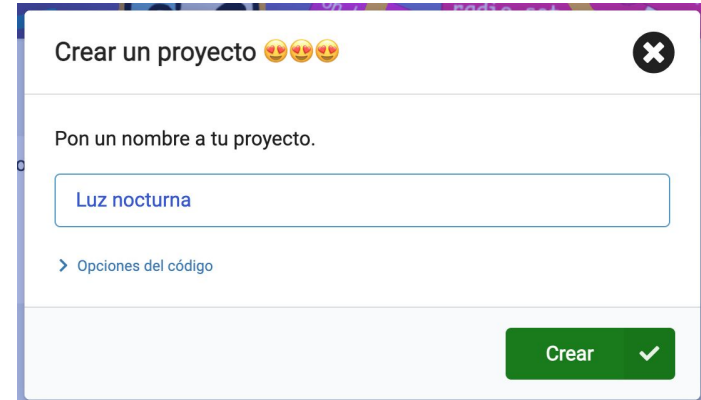
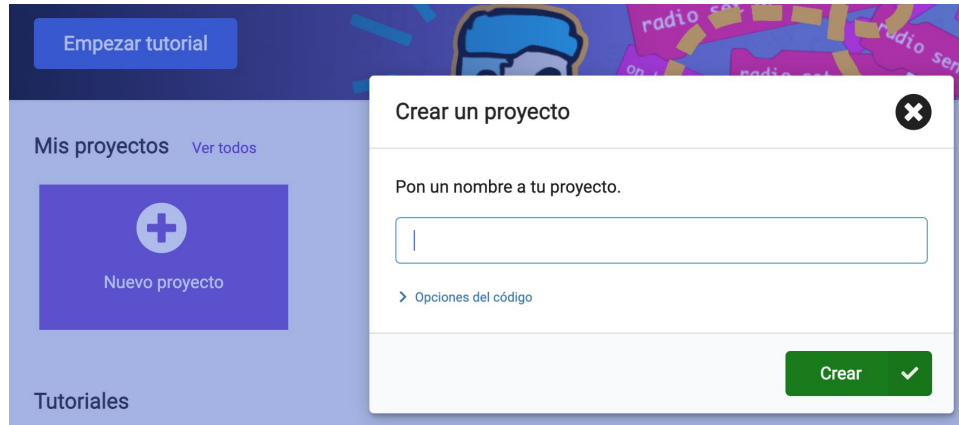
Escribimos en el navegador [makecode.microbit.org](https://makecode.microbit.org)



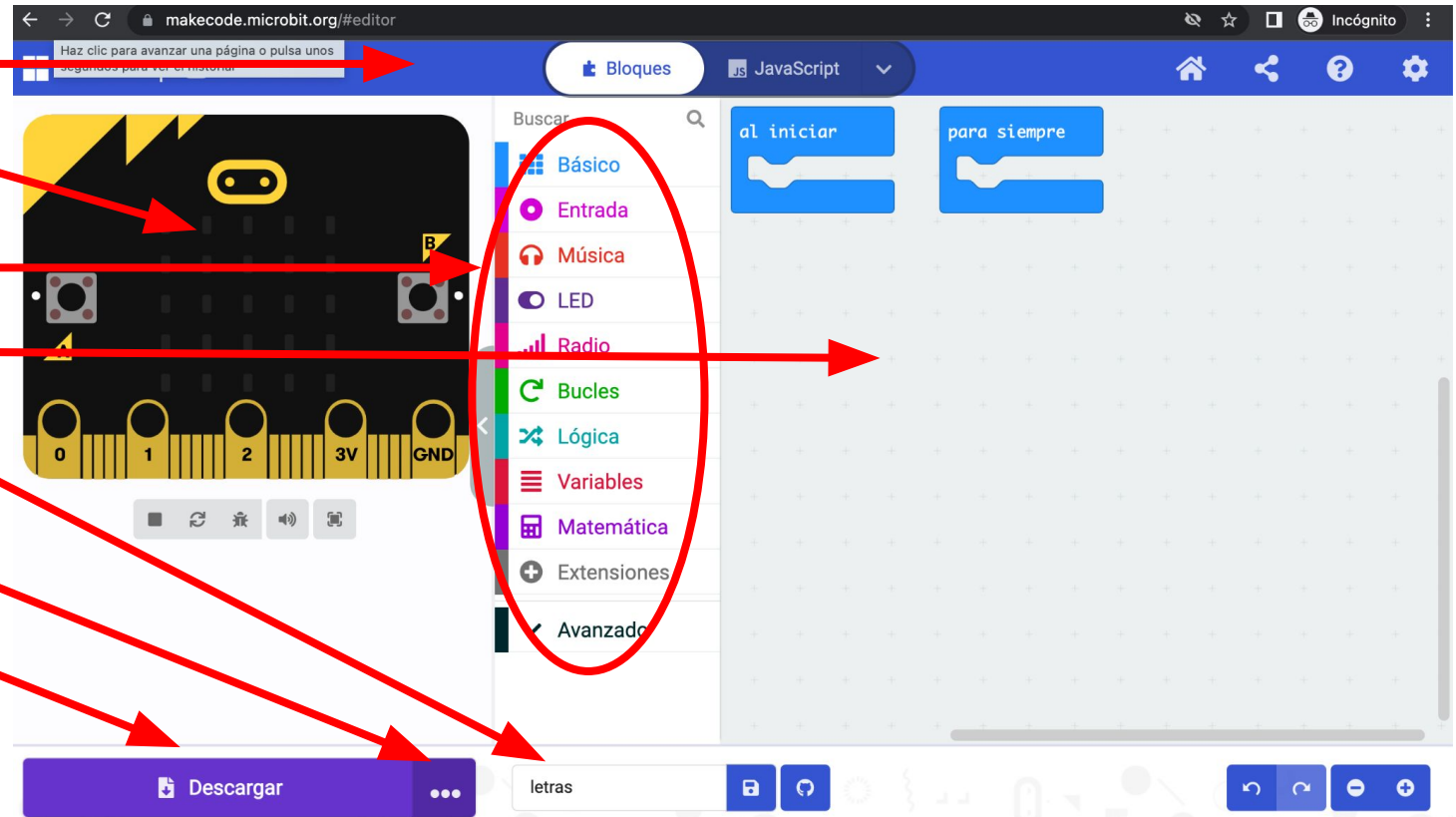
The screenshot shows the Microsoft MakeCode for micro:bit website. The browser's address bar displays `makecode.microbit.org`. The page features a blue header with the Microsoft logo and the text "micro:bit". Below the header is a banner with the text "Behind the MakeCode Hardware" and a button labeled "Ver lista". The main content area is divided into two sections: "Mis proyectos" (My projects) and "Tutoriales" (Tutorials). The "Mis proyectos" section includes a "Nuevo proyecto" (New project) button. The "Tutoriales" section displays five project cards: "Flashing Heart", "Name Tag", "Smiley Buttons", "Dice", and "Love Meter".

# Nuevo proyecto MakeCode

Si pulsamos en “**Nuevo proyecto**” podremos dar un nombre a nuestro proyecto



# Panel de trabajo y simulador de MakeCode



The image shows the MakeCode editor interface with several components labeled in Spanish:

- Opciones:** Points to the top navigation bar containing the 'Bloques' and 'JavaScript' tabs.
- Simulador:** Points to the central area showing a virtual Micro:bit device.
- Bloques:** Points to the 'Entrada' block in the block palette.
- Código:** Points to the code editor area on the right.
- Nombre:** Points to the 'Entrada' block in the block palette.
- Conexión:** Points to the 'Conectar' button in the bottom toolbar.
- Descarga:** Points to the 'Descargar' button in the bottom toolbar.

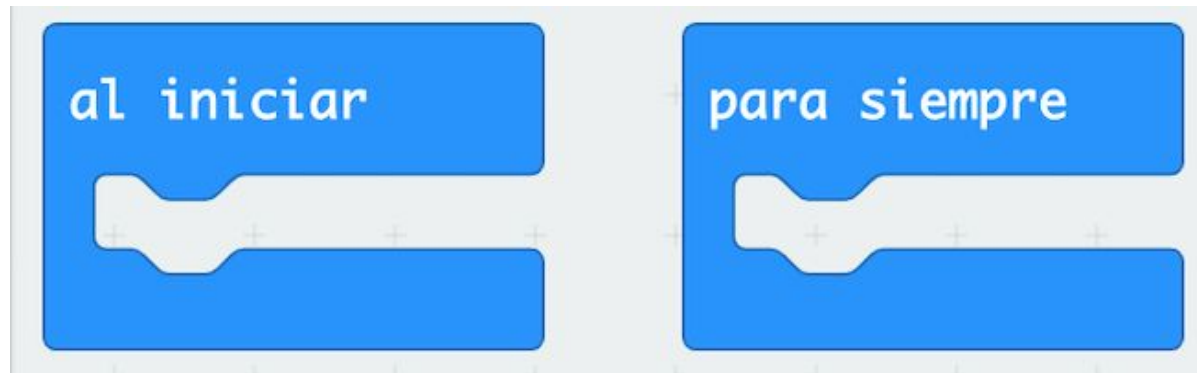
A red circle highlights the block palette, and red arrows indicate the connections between the labels and the interface elements.

# Código por defecto

El código de muestra tiene dos partes:

- Al iniciar - aquí incluiremos el código que se ejecuta una vez
- Para siempre - aquí tenemos el código que se ejecuta repetidamente

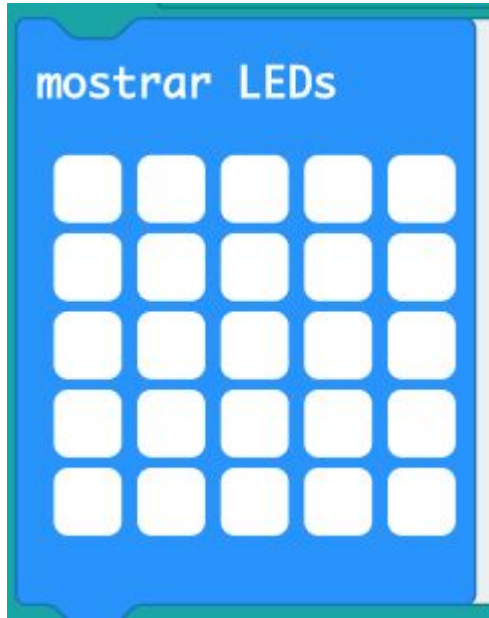
El simulador de la izquierda se activará cuando escribamos un programa



# Pensemos un poco

¿Cómo hacemos **luz** con micro:bit?

Iluminando todos los leds de la pantalla !!

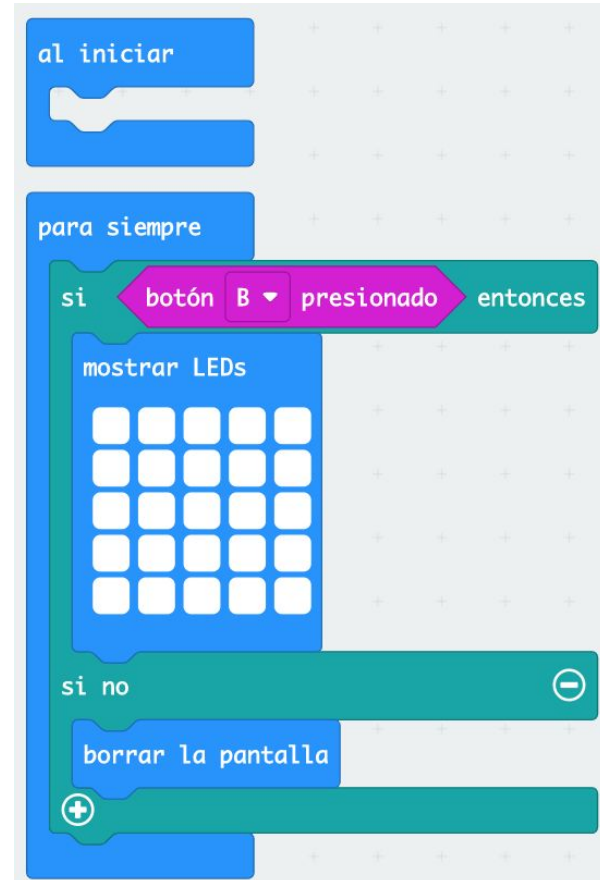


# Linterna

¿Qué programa necesitamos?

Uno muy simple

Al mantener pulsado un botón se enciende la pantalla y al soltarlo se apaga





# Pensemos otra forma

Esta linterna se apaga si la soltamos

¿Cómo podemos hacer que se mantenga encendida?

- encendiendo con un botón y apagando con otro
- encendiendo al pulsar una vez y apagando al volver a pulsar



# Linterna con dos botones

¿Qué programa necesitamos?

La respuesta es simple.

El botón **B** enciende la luz y  
el botón **A** apaga la luz



# Linterna con un botón

Para hacer una linterna con un solo botón:

El botón **B** enciende y apaga la luz

Necesitamos que **micro:bit** recuerde si está encendida o apagada, y esto se consigue con una variable, donde guardaremos el **Estado**

Creemos la variable **Estado**



# Definir una variable

Entre las opciones disponibles, encontramos las **Variables**

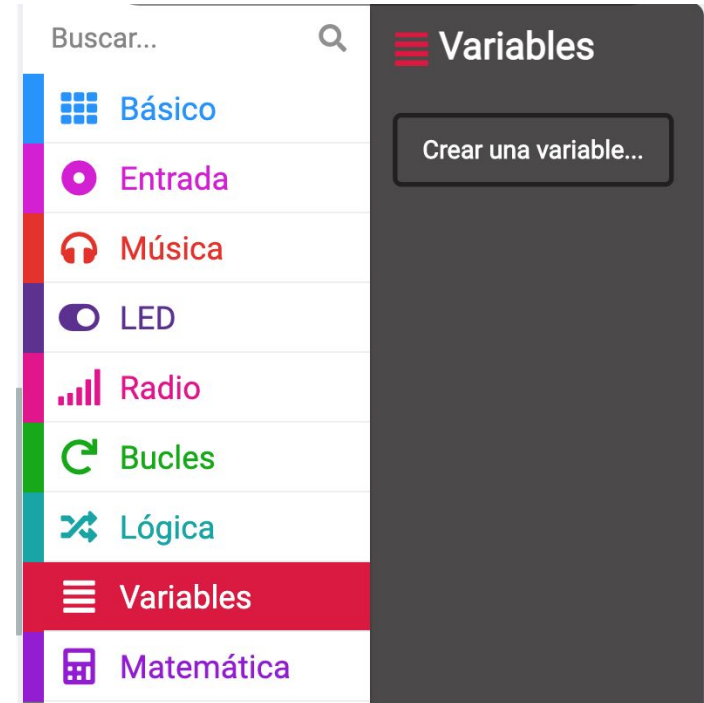
Pulsamos en **Crear una variable** y le damos el nombre **Estado**



Nombre de la nueva variable: ✕

Estado

Aceptar ✓

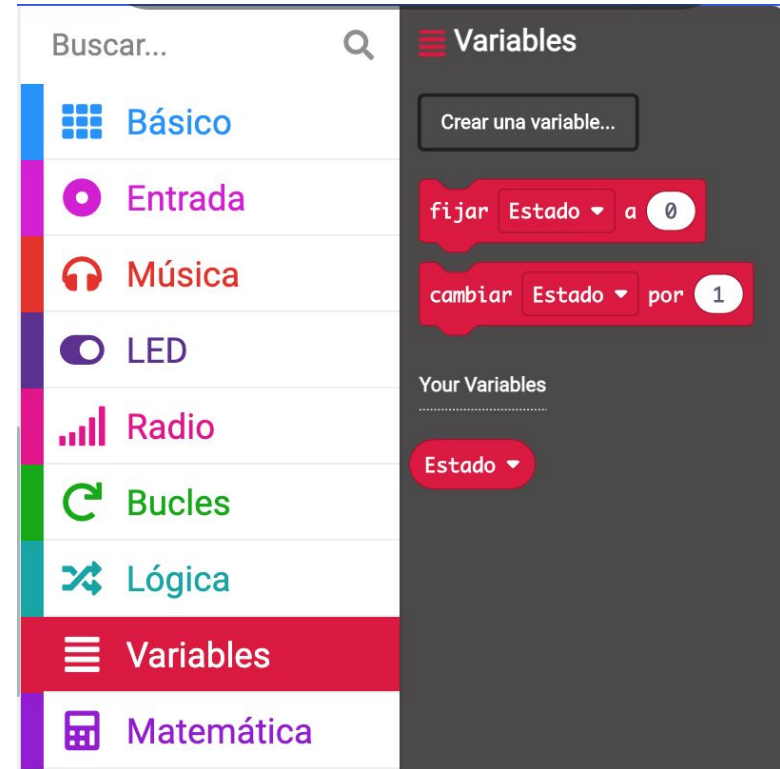


# Opciones de una variable

En el menú de bloques han aparecido tres opciones nuevas, todas con el nombre **Estado**

- La primera nos permite definir el valor
- La segunda nos permite aumentar el valor de **Estado**  
(Cambiar ... por 0) == (Aumentar en 0)
- La tercera nos permite usar el valor de la variable **Estado**

Parece que las variables guardan números



Buscar... 🔍

- Básico
- Entrada
- Música
- LED
- Radio
- Bucles
- Lógica
- Variables**
- Matemática

**Variables**

Crear una variable...

fijar Estado a 0

cambiar Estado por 1

Your Variables

Estado

# Linterna con un botón

Ahora cambiamos completamente la forma de hacer el programa.

- El **botón B** cambia la variable **Estado**
- En **Para siempre** miramos el valor de la variable **Estado** y **encendemos** o **apagamos** la linterna
- Al iniciar ponemos **Estado** a cero



```

al iniciar
  fijar Estado a 0
  
```



```

para siempre
  si Estado = 1 entonces
    mostrar LEDs
  si no
    borrar la pantalla

al presionarse el botón B
  si Estado = 1 entonces
    fijar Estado a 0
  si no
    fijar Estado a 1
  
```

## Pensemos otro uso

¿Cómo podemos hacer una luz que se encienda sola cuando se hace oscuro?

La tarjeta **micro:bit** tiene un **sensor de luz** incorporado. Podemos conocer el valor del sensor de luz con

nivel de luz

El valor del **nivel de luz** va de **0 a 255**



# Pensemos el algoritmo

## DEFINICIÓN

Un algoritmo es una secuencia de pasos bien definidos que resuelven un problema

## NUESTRO ALGORITMO

- Si hay **poca luz** se **encienden** los leds
- Si el nivel de luz es suficiente se **apagan**





# Luz nocturna

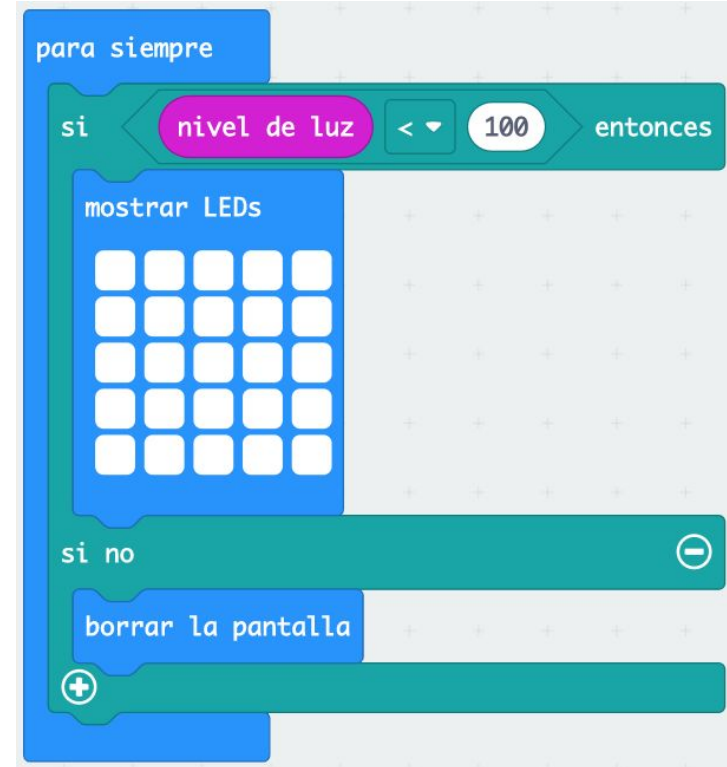
## NO CAMBIES EL PROGRAMA AÚN

El programa puede ser muy simple

Comparamos el **nivel de luz** con un valor (el **umbral de luz** que queramos) y presentamos toda la pantalla iluminada.

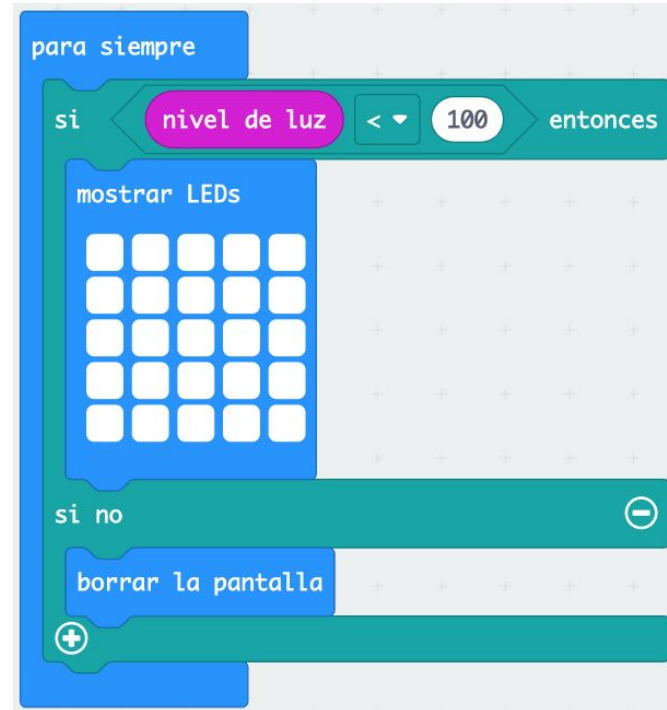
Si hay más luz que el umbral, apagamos la pantalla

**Vamos a combinarlo con el anterior**



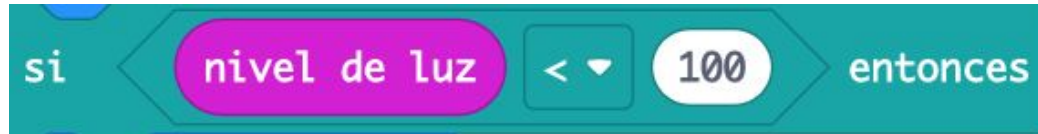
# Luz nocturna

Tenemos dos programas parecidos

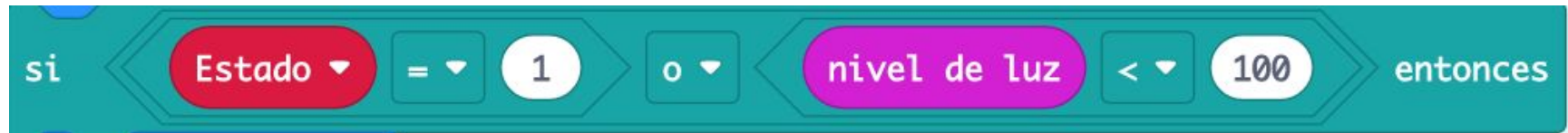
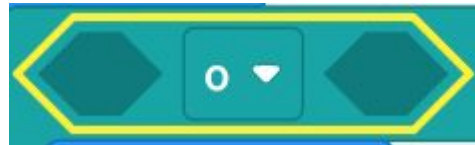


# Combinar comparaciones

Podemos combinar las dos comparaciones



Usaremos la unión de **Lógica**



# Linterna automática

Ya tenemos nuestro programa completo !!

- Podemos encender y apagar la linterna con un botón
- Si no hay luz se enciende sola

```

al iniciar
  fijar Estado a 0
  
```

```

para siempre
  si Estado = 1 o nivel de luz < 100 entonces
    mostrar LEDs
  si no
    borrar la pantalla
  al presionarse el botón B
    si Estado = 1 entonces
      fijar Estado a 0
    si no
      fijar Estado a 1
  
```

# RESUMEN

El programa ya está completo.

Ahora es el momento de copiarlo a nuestra tarjeta **micro:bit** y probarlo de verdad.

Recuerda los pasos:

1. Conectar el micro:bit al ordenador
2. Emparejar el **micro:bit (Connect device)**
3. Descargar el código al **micro:bit**

# Observemos el funcionamiento

Probemos nuestro programa !!

¿Funciona todo bien?

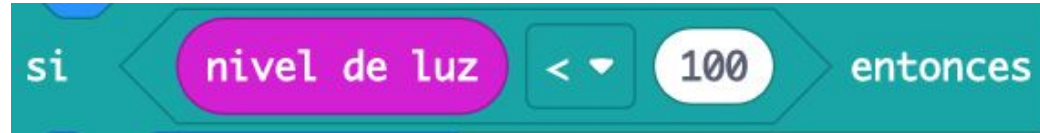
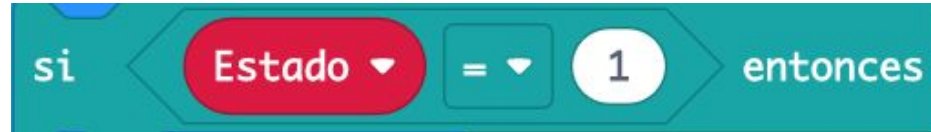
Y por la noche cuando sea oscuro

¿podremos apagar la linterna para dormir?

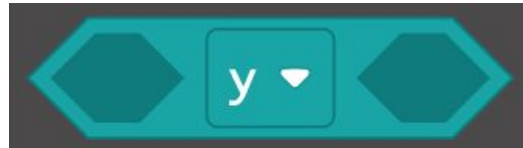


# Combinar comparaciones

Podemos cambiar la forma de unir las dos comparaciones



Usaremos la unión de **Lógica**



# Pensemos un poco

¿Qué hará ahora la linterna?

Hemos escrito que para que se encienda:

- **Estado** tiene que valer 1
- **Nivel de luz** debe ser menor que 100

Como hemos puesto una Y se deben cumplir las dos condiciones a la vez !!

Ahora la luz nocturna sólo funcionará cuando tengamos encendida la linterna y esta sólo se encenderá por la noche !!





# RESUMEN

El programa ya está completo.

Ahora es el momento de copiarlo a nuestra tarjeta **micro:bit** y probarlo de verdad.

Recuerda los pasos:

1. Conectar el micro:bit al ordenador
2. Emparejar el **micro:bit (Connect device)**
3. Descargar el código al **micro:bit**

# Agraïments i Contribucions i Llicència

Les diapositives estan sota el Copyright **2021** © **Steam4all**, i estan disponibles públicament sota una llicència **Creative Commons Attribution 4.0**. amb l'obligació de mantenir aquesta última diapositiva en totes les còpies del document, o una part, per complir amb els requeriments d'atribució de la llicència. Si fas un canvi, ets lliure d'afegir el teu nom i organització a la llista de col·laboradors en aquesta pàgina on siguin publicats els materials.

Han contribuït a la creació d'aquest material

- Joaquin Jimenez Godoy
- Tony Barbosa
- Wouter Molevelt
- Maria Teresa Miras
- Eusebi Calonge

<https://steam4all.eu>

