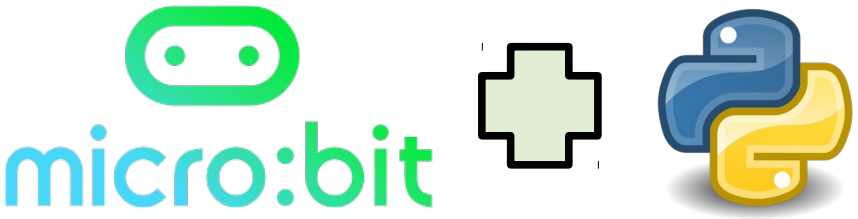
Retos de Micro:bit + Python



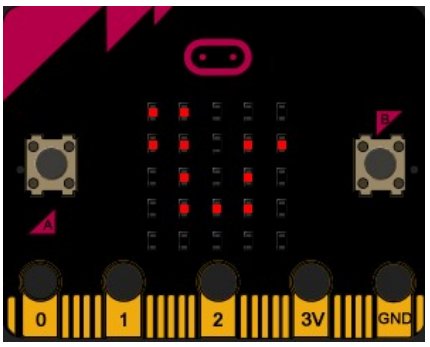
Por: *Pedro Ruiz Fernández*

Versión 17/11/2019

Licencia

Reto 1. Hola Mundo
Es la primera práctica o reto en programación, en nuestro caso mostramos con ayuda de la matriz de leds la frase "hola mundo" en modo texto con desplazamiento (scroll). Al objeto display se le aplica un método o procedimiento (acción) llamada scroll.

```
from microbit import *
display.scroll("Hola Mundo")
```

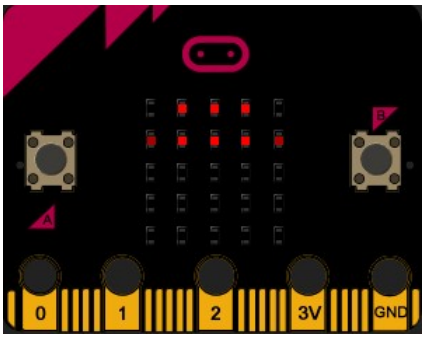


Reto 2. Iconos en display
En este reto al objeto display se le aplica el procedimiento (acción) show que nos permite mostrar imágenes prediseñadas, en esta dirección podéis encontrar otras imágenes prediseñadas.

```
from microbit import *
display.show(Image.SNAKE)
```

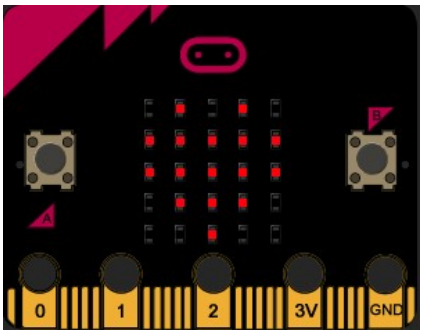
Reto 3. Imágenes DIY
También podemos hacer imágenes en la matriz de leds DIY (Di It Yourself), para ello creamos una variable a la que asociamos una imagen compuesta a base de las diferentes filas (5) compuestas por 5 leds cada una indicando en la posición de cada led su luminosidad de 0 (apagado) a 9 intensidad luminosa máxima. En el ejemplo dibujamos un sombrero.

```
from microbit import *

hat = Image("09990:"
            "59995:"
            "00000:"
            "00000:"
            "00000")

display.show(hat)
```



Reto 4. Animaciones
En este reto creamos un array (matriz) de imágenes, en nuestro caso de corazones de tamaño diferente, para animarlos simulando el pálpito del mismo.

```
from microbit import *
imagenes=[Image.HEART,Image.HEART_SMALL]

while True:
        display.show (imagenes[0])
        sleep(500)
        display.show(imagenes[1])
        sleep(500)
```



Otra solución:

```
from microbit import *
imagenes=[Image.HEART,Image.HEART_SMALL]

while True:
        display.show (imagenes,sleep(500))
```

Reto 5. Animaciones DIY
Este reto trata sobre la animación de un sombrero cayendo, para ello lo que hacemos es dibujar 6 sombreros en 6 posiciones diferentes desapareciendo hacia abajo, por tanto lo último que se verá será la parte alta del sombrero. Para ello además de crear las 6 imágenes, creamos una lista con todos los sombreros, en nuestro caso “hats”, para hacer efectiva la animación a través de la orden display.show (hats, delay=150), lo que hacemos es ejecutar la animación con un tiempo de retardo entre imagen de 150 ms.

```
from microbit import *

hat1 = Image("09990:"
        "59995:"
        "00000:"
        "00000:"
        "00000")

hat2 = Image("00000:"
        "09990:"
        "59995:"
        "00000:"
        "00000")

hat3 = Image("00000:"
        "00000:"
        "09990:"
        "59995:"
        "00000")

hat4 = Image("00000:"
        "00000:"
        "00000:"
        "09990:"
        "59995")

hat5 = Image("00000:"
        "00000:"
        "00000:"
        "00000:"
        "09990")

hat6 = Image("00000:"
        "00000:"
        "00000:"
        "00000:"
        "00000")

hats = [hat1, hat2, hat3, hat4, hat5, hat6]
display.show(hats, sleep(150))
```

Reto 6. Animación para siempre
En este caso se trata de realizar la animación anterior pero introduciendo la misma en un bucle infinito realizado con “while True”.

```
from microbit import *

hat1 = Image("09990:"
        "59995:"
```

```
        "00000:"
        "00000:"
        "00000")

hat2 = Image("00000:"
        "09990:"
        "59995:"
        "00000:"
        "00000")

hat3 = Image("00000:"
        "00000:"
        "09990:"
        "59995:"
        "00000")

hat4 = Image("00000:"
        "00000:"
        "00000:"
        "09990:"
        "59995")

hat5 = Image("00000:"
        "00000:"
        "00000:"
        "00000:"
        "09990")

hat6 = Image("00000:"
        "00000:"
        "00000:"
        "00000:"
        "00000")

hats = [hat1, hat2, hat3, hat4, hat5, hat6]
while True:
        display.show(hats, delay=150)
```

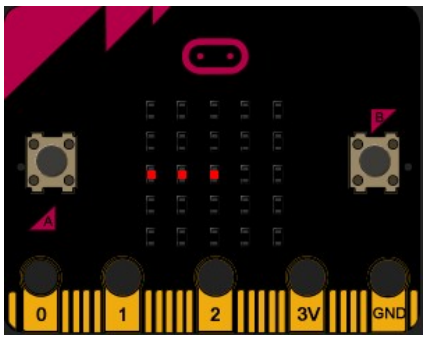
Reto 7. Contador de pulsaciones de botones

En este reto vamos a contar el número de veces que pulsamos el botón a de microbit en un tiempo de 12 segundos, para ello. Usamos la función sleep que deja en modo pausa a microbit durante el número de milisegundos indicado. Además para contar el número de veces que es pulsado el objeto botón a usamos la función buton_a.get_presses().

```
from microbit import *

sleep(12000)
display.scroll(str(button_a.get_presses()))
```

Reto 8. While

Este reto muestra la imagen de Reloj a las 12 mientras el tiempo de funcionamiento de microbit sea inferior a 9 segundos, pasado ese tiempo muestra la imagen Reloj a las 9.

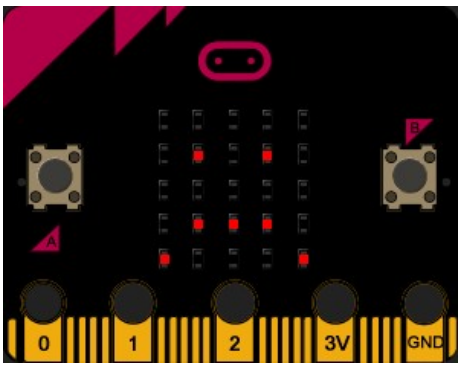



```
from microbit import *

while running_time() < 9000:
```

```
        display.show(Image.CLOCK12)

display.show(Image.CLOCK9)
```

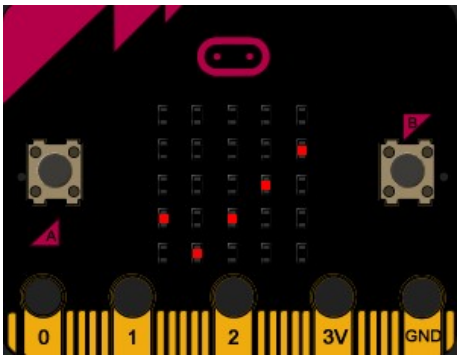
Reto 9. Bucle infinito e if
En este reto si pulsas el botón "A" muestra la imagen SMILE si lo sueltas se muestra la imagen SAD, sale del bucle infinito (while True) cuando pulsas el botón "B".

```
from microbit import *

while True:
        if button_a.is_pressed():
                display.show(Image.SMILE)
        elif button_b.is_pressed():
                break
        else:
                display.show(Image.SAD)
display.clear()
```



Reto 10. Bucle infinito e if con control de variable booleana
Se trata de mostrar una imagen u otra (YES, NO) pulsando un sólo botón, en este caso el botón "B".

```
from microbit import *
on= True

while True:
if button_b.is_pressed():
        on = not on
        sleep (200)
if on==True:
        display.show(Image.YES)
else:
        display.show(Image.NO)
```



Reto 11. Bucle for (repeticiones)
Se trata de realizar la animación del corazón palpitando, alternando las imágenes de corazón normal y pequeño 5 veces usando for.

```
from microbit import *

for i in range (5):
        display.show(Image.HEART)
        sleep (500)
        display.show(Image.HEART_SMALL)
        sleep (500)
display.clear()
```

Reto 12. Bucle for (repeticiones) con matrices de leds, enciende primera fila
Reto consistente en encender la primera fila de leds, secuencialmente de izquierda a derecha cada medio segundo.

```
from microbit import *

for x in range (5):
        display.set_pixel(x,0,9)
        sleep (500)
        display.set_pixel (x,0,0)
```

```
        sleep (500)

display.clear()
```

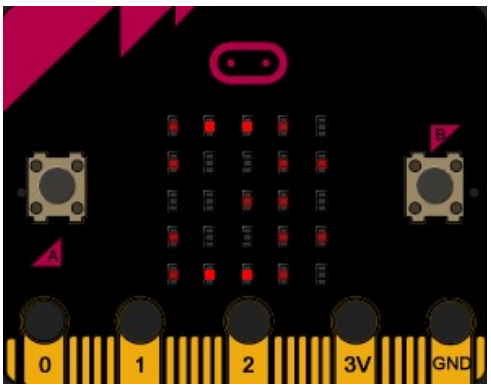
Reto 13. Bucle for (repeticiones) con matrices de leds, enciende todos los leds
Se trata de encender y apagar todos los leds del display secuencialmente de izquierda a derecha y de arriba a abajo cada 50 milisegundos.

```
from microbit import *

for y in range (5):
        for x in range (5):
                display.set_pixel(x,y,9)
                sleep (50)
                display.set_pixel (x,y,0)
                sleep (50)

display.clear()
```

Reto 14. Bucle for (repeticiones) con lista (array)
Reto consistente en encender de la primera fila de leds los leds impares (posiciones 1, 3 y 5), secuencialmente de izquierda a derecha cada 200 milisegundos.

```
from microbit import *

pixels=[0,2,4]

while True:
        for i in range (3):
                display.set_pixel (pixels[i],0,9)
                sleep (200)
                display.set_pixel(pixels[i],0,0)
                sleep (200)
```

Reto 15. Muestra de temperatura.
Reto consistente en mostrar por pantalla la temperatura de microbit cada medio segundo.




```
from microbit import *

while True:
        temp=temperature()
        display.scroll(str(temp)+"ºC")
        sleep(500)
```

Reto 16. Termostato
Si la temperatura de microbit supera el umbral de 32ºC se muestra la imagen YES, si es menor se muestra l aimagen NO.

```
from microbit import *

while True:
        if temperature()>32:
                display.show(Image.YES)
        else:
                display.sow(Image.NO)
```

Reto 17. Encendido gradual de filas de leds en función de luz.
Se trata de encender filas de leds de la pantalla en función de la luz, a más luz más filas encendidas empezando de arriba (menor cantidad de luz) hacia abajo (mayor cantidad de luz).

```
from microbit import *

while True:
        luz=display.read_light_level()
        nivel=round((luz*5)/255)
        #display.scroll(nivel)
        for y in range (nivel):
                for x in range(5):
                        display.set_pixel (x,y,9)
        sleep(500)
        display.clear()
```

¡¡¡ Próximamente retos con: acelerómetro, gestos, radio, pines. !!!