

## Código Pi

### Kit de experimentación

**Cuenta regresiva  
en la Sense HAT**



# Autoridades

**Presidente de la Nación**

Mauricio Macri

**Jefe de Gabinete de Ministros**

Marcos Peña

**Ministro de Educación**

Alejandro Finocchiaro

**Jefe de Gabinete de Asesores**

Javier Mezzamico

**Secretaria de Innovación y Calidad Educativa**

María de las Mercedes Miguel

**Directora Nacional de Innovación Educativa**

María Florencia Ripani

ISBN en trámite

Este material fue producido por el Ministerio de Educación de la Nación, en función de los Núcleos de Aprendizajes Prioritarios, para la utilización de los recursos tecnológicos propuestos en el marco del proyecto Escuelas del Futuro.

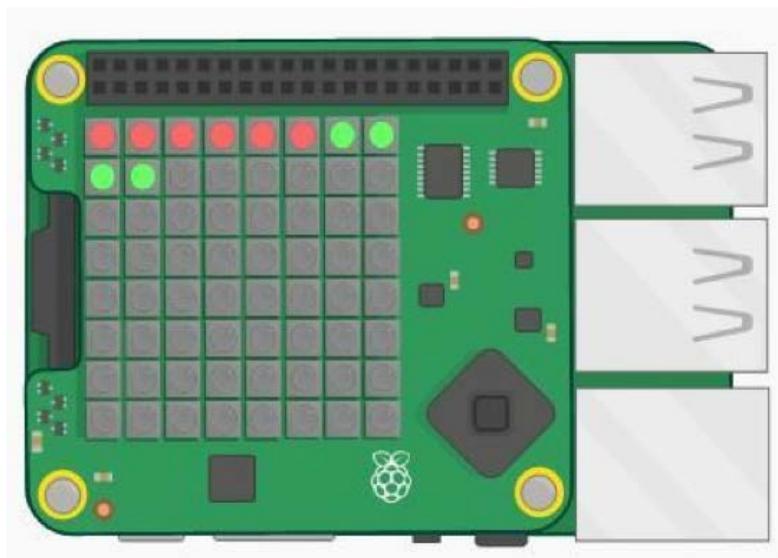
# Índice

Paso 1: Ingresar la cuenta regresiva .....	5
Paso 2: Crear un temporizador de puntos .....	8

# Cuenta regresiva en la Sense HAT

## Introducción

En este proyecto usarás píxeles coloreados en la placa Sense HAT para visualizar un temporizador de cuenta regresiva. Escribirás código en el lenguaje de programación Python.



## Paso 1: Ingresar la cuenta regresiva

Primero vamos a crear la cuenta regresiva de 5 a 0, mediante la visualización de los números, utilizando los píxeles.

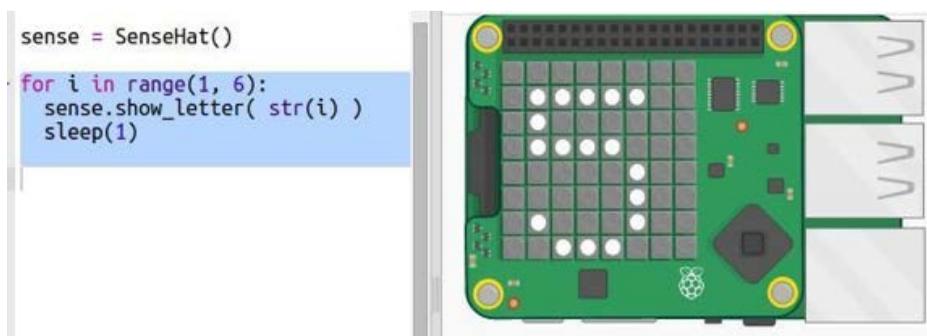
## • Lista de verificación de actividades

Si estás en línea, abrí el sitio web de Trinket para el temporizador de cuenta regresiva: [jumpto.cc/timer-go](http://jumpto.cc/timer-go); si no, copiá este código en el IDE de Python.

```
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()
```

→ Primero, vas a contar hasta 5 porque así es más fácil. Agregá el código resaltado en la parte de abajo de tu programa:



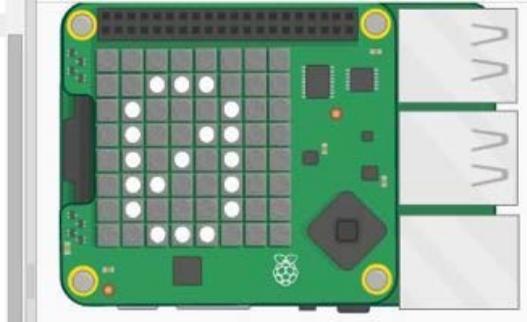
`sense.show_letter()` muestra una sola letra en Sense HAT. No admite números, así que tendrás que usar `str` para cambiar el número.  
`sleep (1)` espera un segundo antes de continuar.

→ En Python, `range(1, 6)` devuelve los números de 1 a 5. Aunque no es necesario contar de a uno:

`range(1, 10, 2)` permitirá agregar números de a dos: por ejemplo, 1, 3, 5, 7, y 9. `range(5, 0, -1)` contará regresivamente quitando un número: por ejemplo, 5, 4, 3, 2, 1

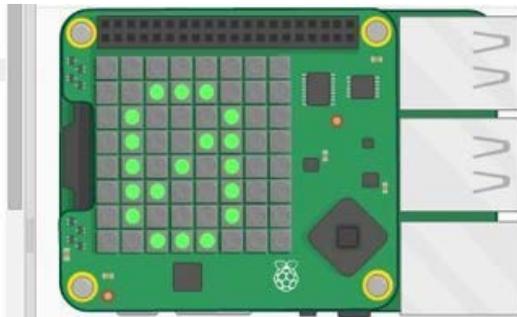
Cambiá el rango en tu código para que cuente regresivamente hasta 0:

```
for i in range(5, -1, -1):  
    sense.show_letter( str(i) )  
    sleep(1)
```

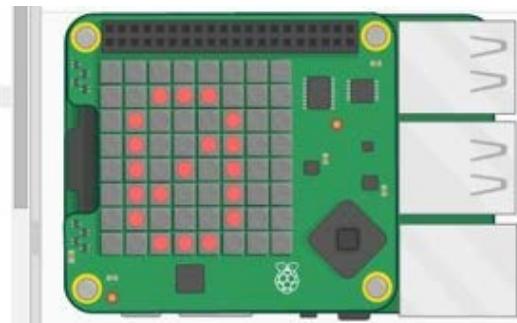


→ El número no tiene por qué ser blanco: Sense HAT permite visualizar muchos colores. Usá colores RGB; probá usando el verde:

```
sense = SenseHat()  
G = [0, 255, 0]  
for i in range(5, -1, -1):  
    sense.show_letter( str(i), G )  
    sleep(1)
```



```
sense = SenseHat()  
G = [0, 255, 0]  
R = [255, 0, 0]  
for i in range(5, -1, -1):  
    sense.show_letter( str(i), R )  
    sleep(1)
```



## Paso 2: Crear un temporizador de puntos

Otra forma de crear un temporizador es cambiar los píxeles de verde a rojo.

- **Lista de verificación de actividades**

Si estás en línea, abrí el sitio web de Trinket para el temporizador de cuenta regresiva: [jumpto.cc/timer-go](https://trinket.io/jumpto.cc/timer-go); si no, copiá este código en el IDE de Phyton.

Si estás en línea, abrí el sitio web de Trinket para temporizador de puntos: [jumpto.cc/dot-timer-go](https://trinket.io/jumpto.cc/dot-timer-go); si no, copiá este código en el IDE de Phyton:

```
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()

G = [0, 255, 0]
R = [255, 0, 0]
```

→ Agregá una variable X para apagar todos los píxeles.

```
G = [0, 255, 0]
R = [255, 0, 0]
X = [0, 0, 0]
```

→ Agregá una variable llamada s para la cantidad de segundos que deseás contar.

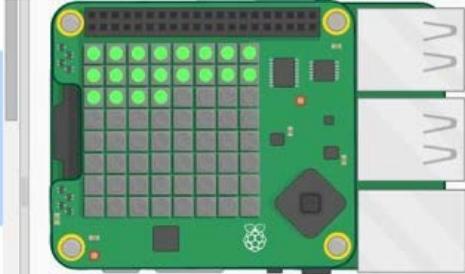
```
G = [0, 255, 0]
R = [255, 0, 0]
X = [0, 0, 0]

s = 20
```

- Podés darle a Sense HAT una lista de 64 (8 x 8) colores para que despliegue, comenzando del extremo superior izquierdo y continuando hacia abajo, una fila por vez.

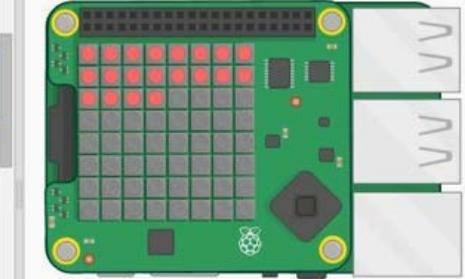
Vamos a crear una lista de colores generando un punto verde para cada segundo que queremos contar y configurando el resto de los 64 píxeles para que se apaguen. La variable `timer` contiene la lista de colores que se visualizarán y empieza vacía:

```
s = 20
timer = []
for i in range(64):
    if i < s:
        timer.append(G)
    else:
        timer.append(X)
sense.set_pixels(timer)
```



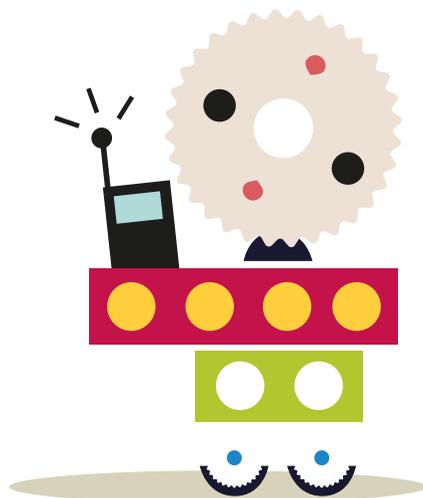
- Ahora ejecutemos la cuenta regresiva convirtiendo un punto en rojo cada segundo:

```
sense.set_pixels(timer)
for i in range(0, s):
    sleep(1)
    timer[i] = R
    sense.set_pixels(timer)
```



- Y qué tal si hacemos aparecer la visualización al final, encendiendo y apagando los píxeles?

```
for i in range(0, 10):
    sense.clear()
    sleep(0.1)
    sense.set_pixels(timer)
    sleep(0.1)
```



**APRENDER  
CONECTADOS**



Ministerio de Educación  
Presidencia de la Nación