

Kit de programación Código Pi

Sense Hat Comandos básicos



Autoridades

Presidente de la Nación

Mauricio Macri

Jefe de Gabinete de Ministros

Marcos Peña

Ministro de Educación, Cultura, Ciencia y Tecnología

Alejandro Finocchiaro

Secretario de Gobierno de Cultura

Pablo Avelluto

**Secretario de Gobierno de Ciencia, Tecnología e
Innovación Productiva**

Lino Barañao

**Titular de la Unidad de Coordinación General
del Ministerio de Educación, Cultura, Ciencia y Tecnología**

Manuel Vidal

Secretaria de Innovación y Calidad Educativa

Mercedes Miguel

Subsecretario de Coordinación Administrativa

Javier Mezzamico

Directora Nacional de Innovación Educativa

María Florencia Ripani

ISBN en trámite



Este material fue producido por el Ministerio de Educación, Cultura, Ciencia y Tecnología en base a contenidos provistos sin cargo por la Fundación Raspberry Pi mediante licencias Creative Commons y han sido desarrollados en función de los Núcleos de Aprendizajes Prioritarios de educación digital, programación y robótica y los recursos tecnológicos propuestos en el marco del Plan Aprender Conectados.

Índice

Destellos aleatorios en Sense HAT.....	5
Configurar píxeles.....	7
Agregar un bucle.....	11
¿Qué sigue?.....	13

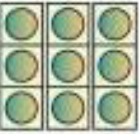

Sense HAT –Comandos básicos

Hoja de referencia para Python

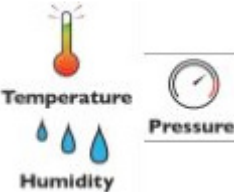
Para incorporar funcionalidad **Sense HAT** a nuestros programas Python, agregamos las siguientes líneas para importar la biblioteca, a la biblioteca Sense HAT:


```
from sense_hat import SenseHat sense = SenseHat()
```

De ahí en más, podremos usar cualquiera de las funciones definidas de la biblioteca Sense HAT.

 <p>LED Matrix</p>	<code>sense.set_pixel(0, 0, 255, 0, 0)</code>	Configura el led del extremo superior izquierdo con el color rojo.
	<code>sense.show_letter("J", 0, 0, 255)</code>	Muestra la letra "J" en color azul en la pantalla.
	<code>sense.show_message("msg", text_colour=[0, 255, 0])</code>	Muestra el mensaje "msg" en verde en la matriz.
	<code>sense.load_image("creeper.png", redraw=True)</code>	Carga una imagen 8x8 "creeper.png" y la muestra.
	<code>sense.clear()</code>	Elimina el led y apaga todos los led.
	<code>sense.set_rotation(r=0)</code>	Configura la rotación de la matriz led.
	<code>sense.set_pixels(pixelList)</code>	Usa una lista de píxeles para hacer un dibujo, cada elemento es una lista [R, G, B].
 <p>Movement</p>	<code>yaw, pitch, roll = sense.get_orientation().values()</code>	Obtiene los datos de orientación y guarda los valores como eje vertical, eje lateral, eje longitudinal .
	<code>m_x, m_y, m_z = sense.get_compass_raw().values()</code>	Obtiene los datos de la brújula y los guarda como m_x, m_y, m_z .
	<code>x, y, z = sense.get_accelerometer_raw().values()</code>	Obtiene los datos del acelerómetro y los guarda como x, y, z .
	<code>g_x, g_y, g_z = sense.get_gyroscope_raw().values()</code>	Obtiene los datos de orientación y los guarda como g_x, g_y, g_z .

APRENDER CONECTADOS

	<code>t = sense.get_temperature_from_humidity()</code>	Usa el sensor de humedad para obtener la temperatura y la guarda como t .
	<code>t = sense.get_temperature_from_pressure()</code>	Usa el sensor de presión para obtener la temperatura y la guarda como t .
	<code>h = sense.get_humidity()</code>	Mide la humedad y la guarda como h .
	<code>p = sense.get_pressure()</code>	Mide la presión y la guarda como p .

	<p>Hay varias maneras de capturar la entrada desde el <i>joystick</i>. Podríamos usar la biblioteca Pygame o la biblioteca Curses. Sin embargo, para este ejemplo vamos a usar el sistema evdev, que tendremos que instalar usando “<code>sudo pip3 install evdev</code>”.</p>	
	<pre>from evdev import InputDevice, ecodes, list_device from select import select devices = [InputDevice(fn) for fn in list_device()] for dev in devices: if dev.name == "Raspberry Pi Sense HAT Joystick": js = dev while True: r, w, x = select([dev.fd], [], [1]), 0.0 for fd in r: for event in dev.read(): if event.type == ecodes.EV_KEY and event.value == 1: if event.code == ecodes.KEY_UP: print("up") elif event.code == ecodes.KEY_LEFT: print("left") elif event.code == ecodes.KEY_RIGHT: print("right") elif event.code == ecodes.KEY_DOWN: print("down") else: print("enter")</pre>	<p>El código de la izquierda busca los dispositivos de entrada disponibles y encuentra el <i>joystick</i> Sense HAT.</p> <p>Luego verifica constantemente el dispositivo <i>joystick</i> y crea una llamada de lista de eventos r.</p> <p>Para cada evento en la lista, verifica si fue un evento de teclado.</p> <p>Luego compara el código de la tecla con los valores para arriba, abajo, izquierda y derecha e imprime el mensaje correspondiente.</p>

Desplazamiento de mensaje

```
from sense_hat import SenseHat

sense = SenseHat()

while True:
    sense.show_message("Spaaaaaaace!!", scroll_speed=0.05, text_colour=[255,255,0], back_colour=[0,0,255])
```

Sensores ambientales

```
from sense_hat import SenseHat

sense = SenseHat()

while True:
    t = sense.get_temperature()
    p = sense.get_pressure()
    h = sense.get_humidity()

    t = round(t,1)
    p = round(p,1)
    h = round(h,1)

    msg = "Temp = %s, Pressure=%s, Humidity=%s" % (t,p,h)

    sense.show_message(msg, scroll_speed=0.05)
```

Rotación de la letra "J"

```
from sense_hat import SenseHat
import time
sense = SenseHat()

sense.show_letter("J")

while True:

    x, y, z = sense.get_accelerometer_raw().values()

    x = round(x, 0)
    y = round(y, 0)

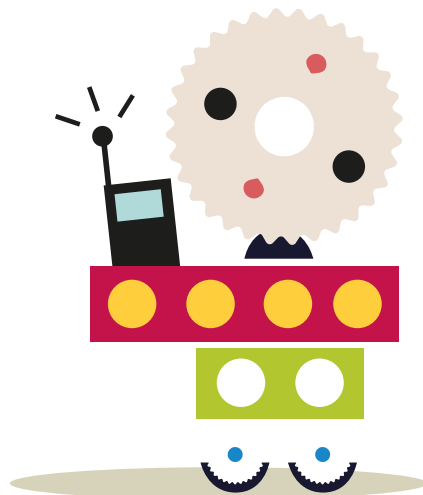
    if x == -1:
        sense.set_rotation(180)
    elif y == -1:
        sense.set_rotation(90)
    elif y == 1:
        sense.set_rotation(270)
    else:
        sense.set_rotation(0)

    time.sleep(0.1)
```

APRENDER CONECTADOS

```
Juego de velocidad de reacción
from sense_hat import SenseHat
import time
import random
sense = SenseHat()
# set up the colours (white, green, red, empty)
w = [150,150,150]
g = [0,255,0] r = [255,0,0] e = [0,0,0]
# create three different coloured arrows arrow =
[e,e,e,w,w,e,e,e,
e,e,w,w,w,w,e,e, e,w,e,w,w,e,w,e,
w,e,e,w,w,e,e,w, e,e,e,w,w,e,e,e,
e,e,e,w,w,e,e,e, e,e,e,w,w,e,e,e,
e,e,e,w,w,e,e,e]
arrow_red = [e,e,e,r,r,e,e,e, e,e,r,r,r,r,e,e,
e,r,e,r,r,e,r,e, r,e,e,r,r,e,e,r, e,e,e,r,r,e,e,e,
e,e,e,r,r,e,e,e, e,e,e,r,r,e,e,e, e,e,e,r,r,e,e,e]
arrow_green = [e,e,e,g,g,e,e,e, e,e,g,g,g,g,e,e,
e,g,e,g,g,e,g,e, g,e,e,g,g,e,e,g, e,e,e,g,g,e,e,e,
e,e,e,g,g,e,e,e, e,e,e,g,g,e,e,e, e,e,e,g,g,e,e,e]
```

```
pause = 3
score = 0
angle = 0
play =
True
sense.show_message("Keep the arrow pointing up", text_colour=[100,100,100])
while play == True:
    last_angle = angle
    while angle == last_angle:
        angle = random.choice([0,90,180,270])
        sense.set_rotation(angle)
        sense.set_pixels(arrow)
        time.sleep(pause)
    x, y, z = sense.get_accelerometer_raw().values()
    x = round(x, 0)
    y = round(y, 0)
    if x == -1 and angle == 180:
        sense.set_pixels(arrow_green)
        score = score + 1
    elif x == 1 and angle == 0:
        sense.set_pixels(arrow_green)
        score = score + 1
    elif y == -1 and angle == 90:
        sense.set_pixels(arrow_green)
        score = score + 1
    elif y == 1 and angle == 270:
        sense.set_pixels(arrow_green)
        score = score + 1
    else: sense.set_pixels(arrow_red)
    play = False
    pause = pause *
    0.95
    time.sleep(0.5)
    msg = "Your score was %s" % (score)
    sense.show_message(msg, scroll_speed=0.05, text_colour=[100,100,100])
```

**APRENDER
CONECTADOS**



Ministerio de Educación,
Cultura, Ciencia y Tecnología
Presidencia de la Nación