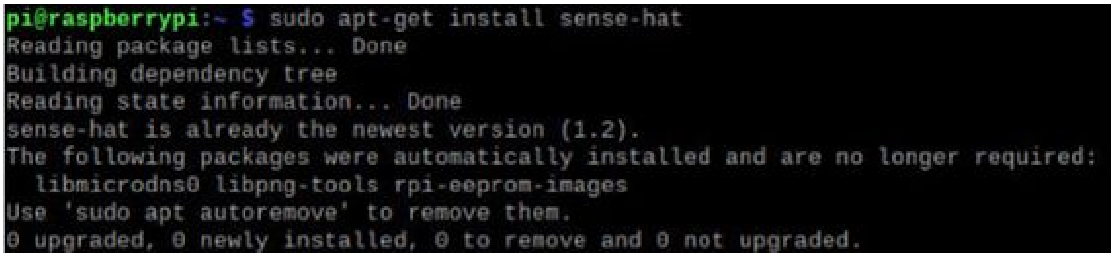
How to use the Sense HAT?

1. Open **terminal**. Install the Sense HAT package by entering the following command.

```
sudo apt-get install sense-hat
```

   After running this command, you should get the result similar to the following:



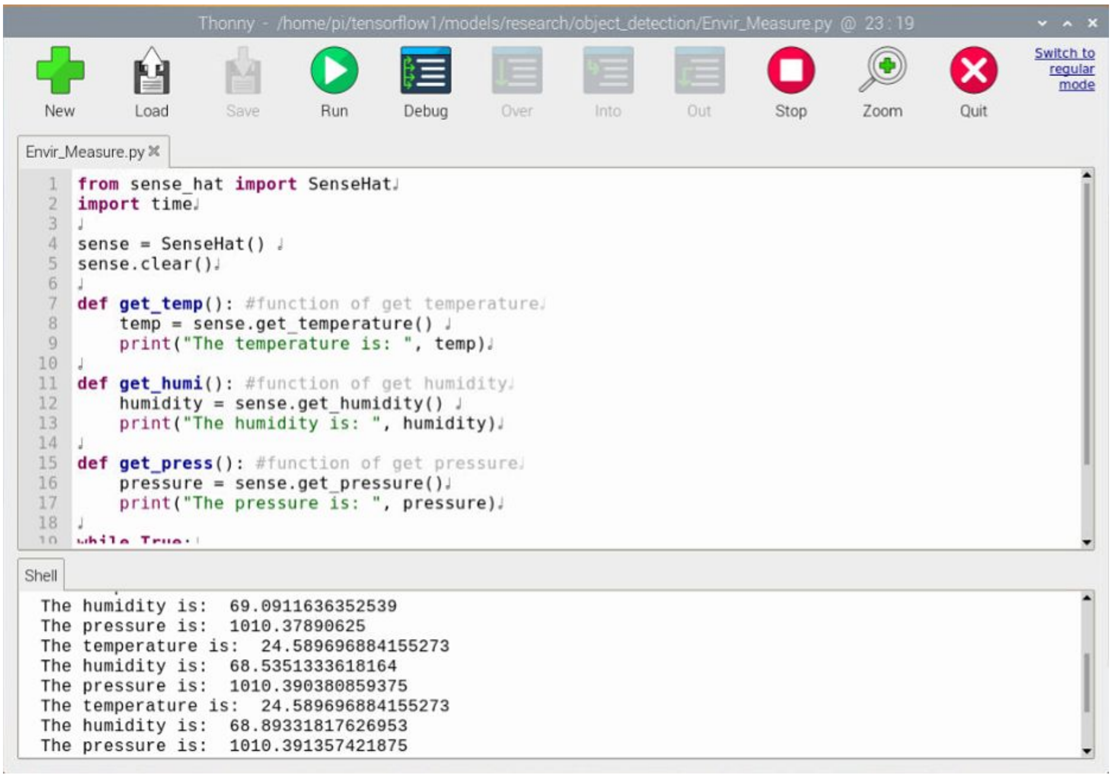
```
pi@raspberrypi:~ $ sudo apt-get install sense-hat
Reading package lists... Done
Building dependency tree
Reading state information... Done
sense-hat is already the newest version (1.2).
The following packages were automatically installed and are no longer required:
  libmicrodns0 libpng-tools rpi-eeprom-images
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

2. Try to measure the surrounding environment of the Raspberry Pi using environmental sensors. Open **Thonny Python IDE** and copy the following program. Save the program and name it `Envir_Measure.py`. Click “**Run**”.

```
from sense_hat import SenseHat
import time

sense = SenseHat()
sense.clear()

def get_temp(): #function of get temperature
    temp = sense.get_temperature()
    print("The temperature is: ", temp)

def get_humi(): #function of get humidity
    humidity = sense.get_humidity()
    print("The humidity is: ", humidity)

def get_press(): #function of get pressure
    pressure = sense.get_pressure()
    print("The pressure is: ", pressure)

while True:
    get_temp()
    get_humi()
    get_press()
    time.sleep(5)
```

   This program measure the temperature, humidity and pressure using a set of environmental sensors on the Sense HAT. Then, print out the readings every 5 seconds.

After running this command, you should get the result similar to the following:


Thonny - /home/pi/tensorflow1/models/research/object_detection/Envir_Measure.py @ 23 : 19
New
Load
Save
Run
Debug
Over
Into
Out
Stop
Zoom
Quit
Switch to regular mode
Envir_Measure.py
Shell

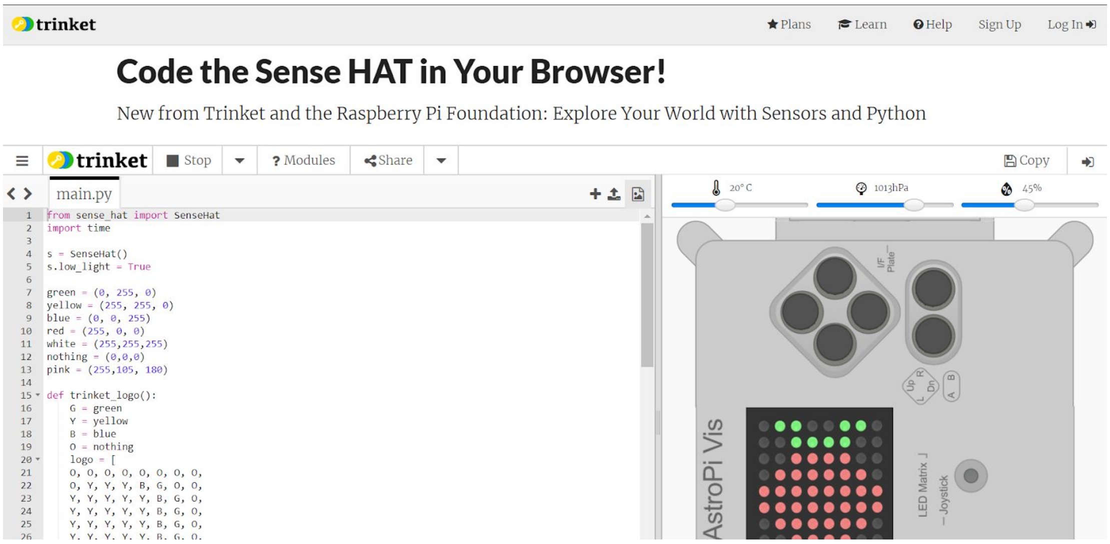
```
from sense_hat import SenseHat
import time

sense = SenseHat() 
sense.clear()

def get_temp(): #function of get temperature
    temp = sense.get_temperature() 
    print("The temperature is: ", temp)

def get_humi(): #function of get humidity
    humidity = sense.get_humidity() 
    print("The humidity is: ", humidity)

def get_press(): #function of get pressure
    pressure = sense.get_pressure()
    print("The pressure is: ", pressure)

while True:
```

```
The humidity is:  69.0911636352539
The pressure is:  1010.37890625
The temperature is:  24.589696884155273
The humidity is:  68.5351333618164
The pressure is:  1010.390380859375
The temperature is:  24.589696884155273
The humidity is:  68.89331817626953
The pressure is:  1010.391357421875
```

Click “**Stop**” if you want to end the measuring.

3. Since you are taking online classes and could not interact with the Sense HAT. Hence, the examples below will be executed using **online emulator**.

4. Open the online emulator from the following website: https://trinket.io/sense-hat. You should be able to see the following.

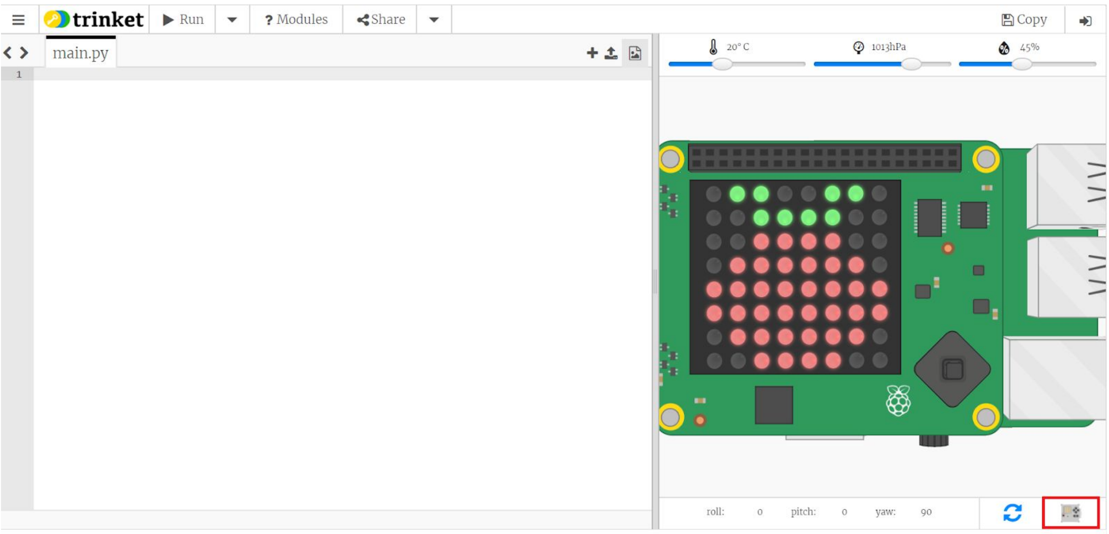


Code the Sense HAT in Your Browser!

New from Trinket and the Raspberry Pi Foundation: Explore Your World with Sensors and Python

```
from sense_hat import SenseHat
import time

s = SenseHat()
s.low_light = True

green = (0, 255, 0)
yellow = (255, 255, 0)
blue = (0, 0, 255)
red = (255, 0, 0)
white = (255,255,255)
nothing = (0,0,0)
pink = (255,105, 180)

def trinket_logo():
    G = green
    Y = yellow
    B = blue
    O = nothing
    logo = [
    O, O, O, O, O, O, O, O,
    O, Y, Y, Y, B, G, O, O,
    Y, Y, Y, Y, Y, B, G, O,
    Y, Y, Y, Y, Y, B, G, O,
    Y, Y, Y, Y, Y, B, G, O,
    Y, Y, Y, Y, Y, B, G, O,
```

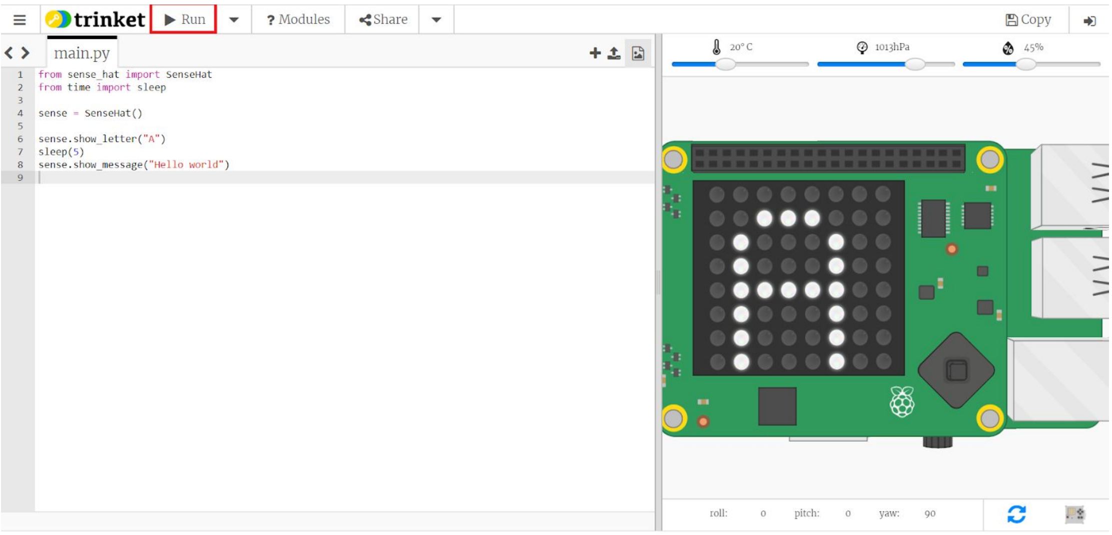
Delete all the code in the `main.py` and press the button in the **lower right corner** to switch the model.



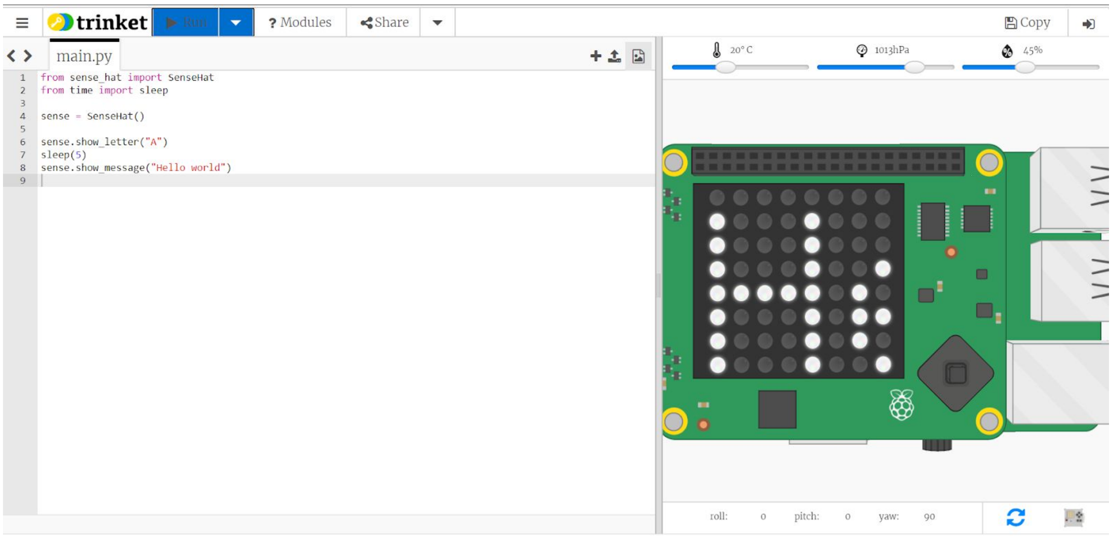
5. Try to display a letter and message using 8x8 RGB LED Matrix. Copy the following program to the `main.py`. Then, click “**Run**” on the top navigation bar.

```
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()

sense.show_letter("A")
sleep(5)
sense.show_message("Hello world")
```

You should be able to see the following result:



```
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()

sense.show_letter("A")
sleep(5)
sense.show_message("Hello world")
```

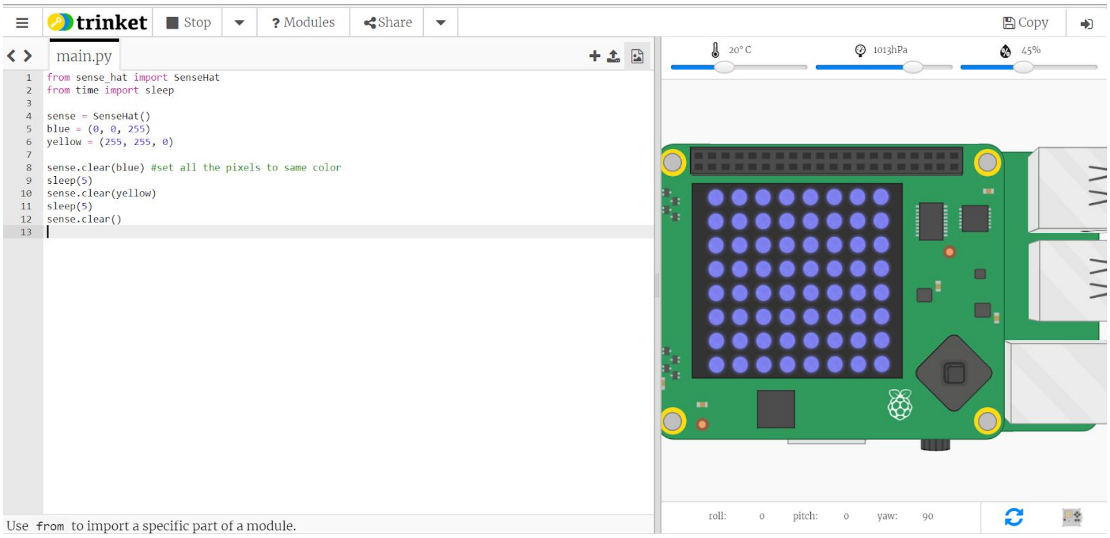


```
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()

sense.show_letter("A")
sleep(5)
sense.show_message("Hello world")
```

6. Try to display a color with all pixels using 8x8 RGB LED Matrix. Copy the following program to the `main.py`. Click “**Run**”.

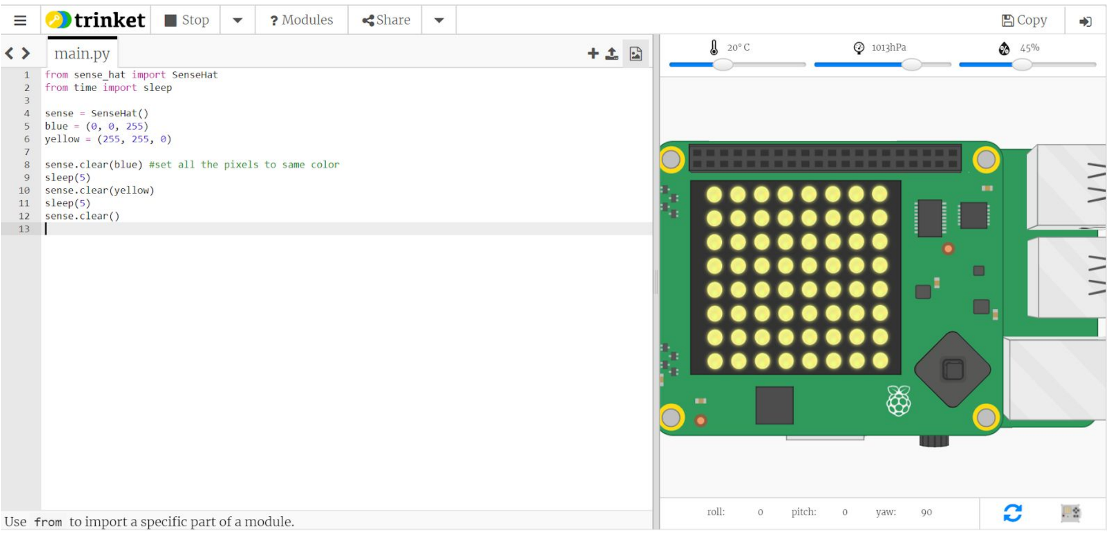
```
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()
blue = (0, 0, 255)
yellow = (255, 255, 0)

sense.clear(blue) #set all the pixels to same color
sleep(5)
sense.clear(yellow)
sleep(5)
sense.clear()
```

You should be able to see the following result:



```
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()
blue = (0, 0, 255)
yellow = (255, 255, 0)

sense.clear(blue) #set all the pixels to same color
sleep(5)
sense.clear(yellow)
sleep(5)
sense.clear()
```

Use `from` to import a specific part of a module.



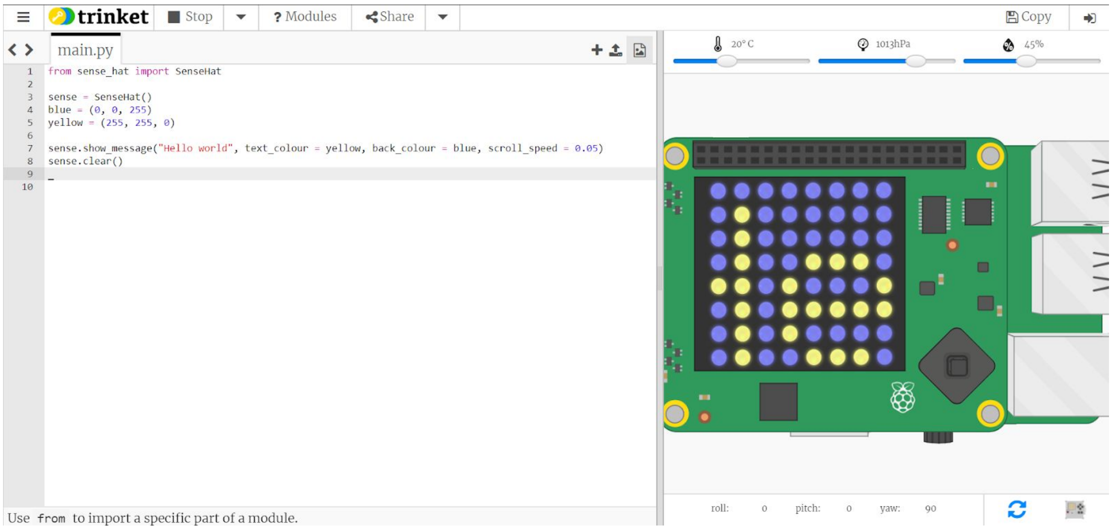
```
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()
blue = (0, 0, 255)
yellow = (255, 255, 0)

sense.clear(blue) #set all the pixels to same color
sleep(5)
sense.clear(yellow)
sleep(5)
sense.clear()
```

Use `from` to import a specific part of a module.

7. Try to display message with different feature using 8x8 RGB LED Matrix. Copy the following program to the `main.py`. Click “**Run**”.

```
from sense_hat import SenseHat

sense = SenseHat()
blue = (0, 0, 255)
yellow = (255, 255, 0)

sense.show_message("Hello world", text_colour = yellow,
back_colour = blue, scroll_speed = 0.05)
sense.clear()
```

You should be able to see the following result:



```
from sense_hat import SenseHat

sense = SenseHat()
blue = (0, 0, 255)
yellow = (255, 255, 0)

sense.show_message("Hello world", text_colour = yellow, back_colour = blue, scroll_speed = 0.05)
sense.clear()
_
```

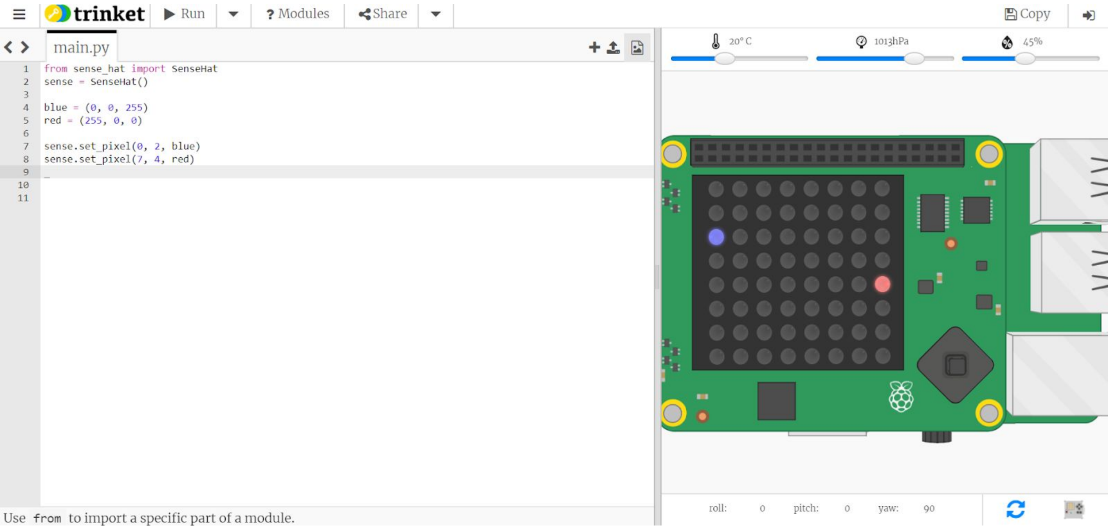
Use `from` to import a specific part of a module.

You can try to change the **message**, **text color**, **background color** and **speed**. The default value of speed is 0.1. The bigger the number, the lower the speed.

8. Try to set the individual pixel with 8x8 RGB LED Matrix. Copy the following program to the `main.py`. Click “**Run**”.

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

blue = (0, 0, 255)
red = (255, 0, 0)

sense.set_pixel(0, 2, blue)
sense.set_pixel(7, 4, red)
```

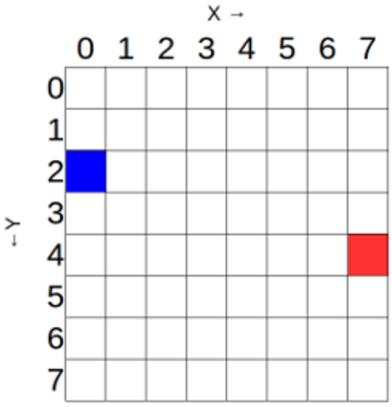
You should be able to see the following result:



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

blue = (0, 0, 255)
red = (255, 0, 0)

sense.set_pixel(0, 2, blue)
sense.set_pixel(7, 4, red)
```

Use `from` to import a specific part of a module.

Noted that the Sense HAT LED matrix uses a coordinate system with an x- and a y-axis. The numbering of both axes begins at 0 (not 1) in the top left-hand corner. Each LED can be used as one pixel of an image, and it can be addressed using an x, y notation.

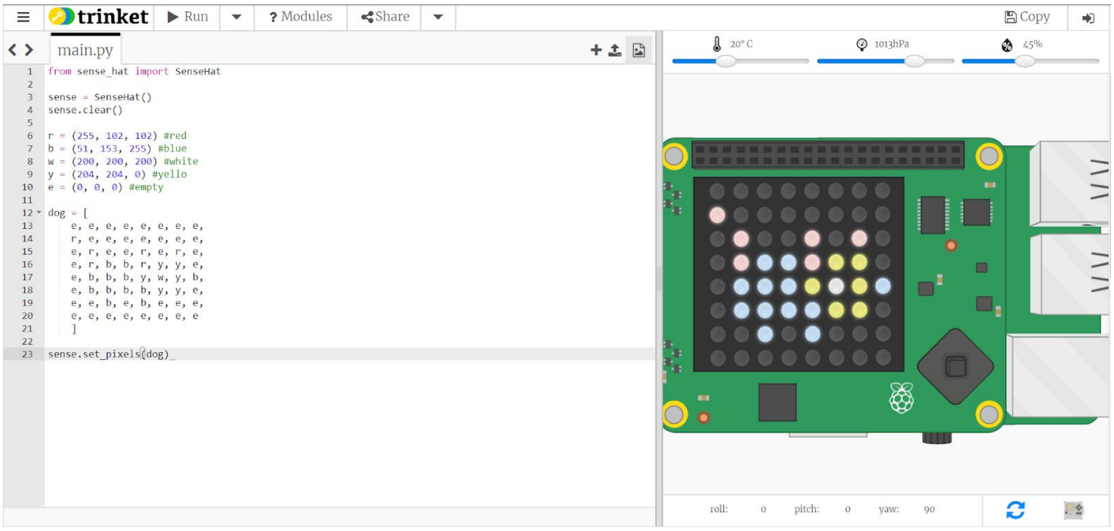


	0	1	2	3	4	5	6	7
0								
1								
2	🟦							
3								
4								🟥
5								
6								
7								

9. Try to design and draw the image using 8x8 RGB LED Matrix. Copy the following program to the `main.py`. Click “**Run**”.

```
from sense_hat import SenseHat

sense = SenseHat()
sense.clear()

r = (255, 102, 102) #red
b = (51, 153, 255) #blue
w = (200, 200, 200) #white
y = (204, 204, 0) #yellow
e = (0, 0, 0) #empty

dog = [
    e, e, e, e, e, e, e, e,
    r, e, e, e, e, e, e, e,
    e, r, e, e, r, e, r, e,
    e, r, b, b, r, y, y, e,
    e, b, b, b, y, w, y, b,
    e, b, b, b, b, y, y, e,
    e, e, b, e, b, e, e, e,
    e, e, e, e, e, e, e, e
    ]

sense.set_pixels(dog)
```

You should be able to see the following result:



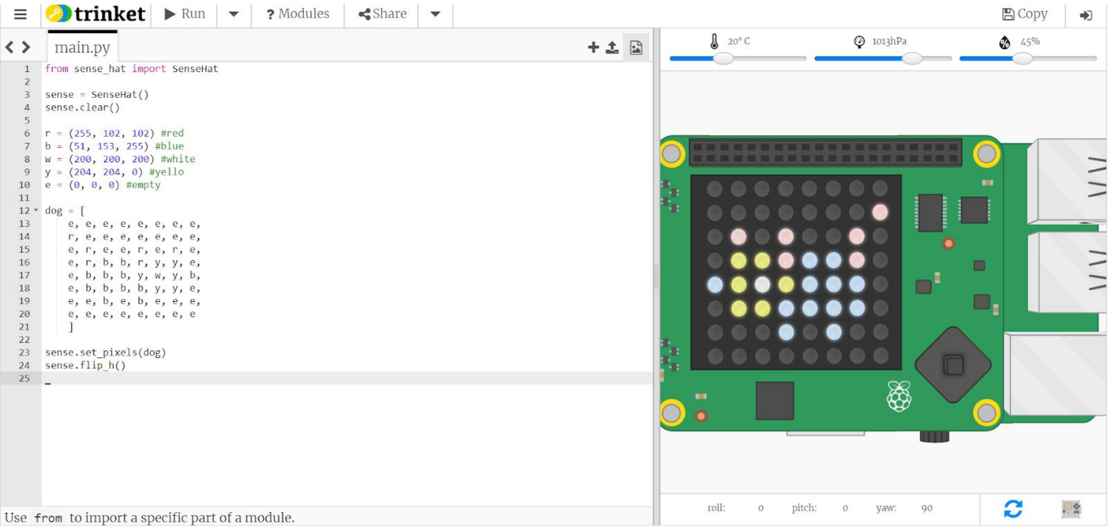
```
from sense_hat import SenseHat

sense = SenseHat()
sense.clear()

r = (255, 102, 102) #red
b = (51, 153, 255) #blue
w = (200, 200, 200) #white
y = (204, 204, 0) #yello
e = (0, 0, 0) #empty

dog = [
    e, e, e, e, e, e, e, e,
    r, e, e, e, e, e, e, e,
    e, r, e, e, r, e, r, e,
    e, r, b, b, r, y, y, e,
    e, b, b, b, y, w, y, b,
    e, b, b, b, b, y, y, e,
    e, e, b, e, b, e, e, e,
    e, e, e, e, e, e, e, e
    ]

sense.set_pixels(dog)
```

After trying this program, you can try to **design a picture** with 8x8 pixels and display them on the LED screen.

10. Try the function flip and rotate of the 8x8 RGB LED Matrix. Copy the following program to the `main.py`. Click "**Run**".

```
from sense_hat import SenseHat

sense = SenseHat()
sense.clear()

r = (255, 102, 102) #red
b = (51, 153, 255) #blue
w = (200, 200, 200) #white
y = (204, 204, 0) #yellow
e = (0, 0, 0) #empty

dog = [
    e, e, e, e, e, e, e, e,
    r, e, e, e, e, e, e, e,
    e, r, e, e, r, e, r, e,
    e, r, b, b, r, y, y, e,
    e, b, b, b, y, w, y, b,
    e, b, b, b, b, y, y, e,
    e, e, b, e, b, e, e, e,
    e, e, e, e, e, e, e, e
    ]

sense.set_pixels(dog)
sense.flip_h()
```

You should be able to see the following result:



```
from sense_hat import SenseHat

sense = SenseHat()
sense.clear()

r = (255, 102, 102) #red
b = (51, 153, 255) #blue
w = (200, 200, 200) #white
y = (204, 204, 0) #yello
e = (0, 0, 0) #empty

dog = [
    e, e, e, e, e, e, e, e,
    r, e, e, e, e, e, e, e,
    e, r, e, e, r, e, r, e,
    e, r, b, b, r, y, y, e,
    e, b, b, b, y, w, y, b,
    e, b, b, b, b, y, y, e,
    e, e, b, e, b, e, e, e,
    e, e, e, e, e, e, e, e
    ]

sense.set_pixels(dog)
sense.flip_h()
```

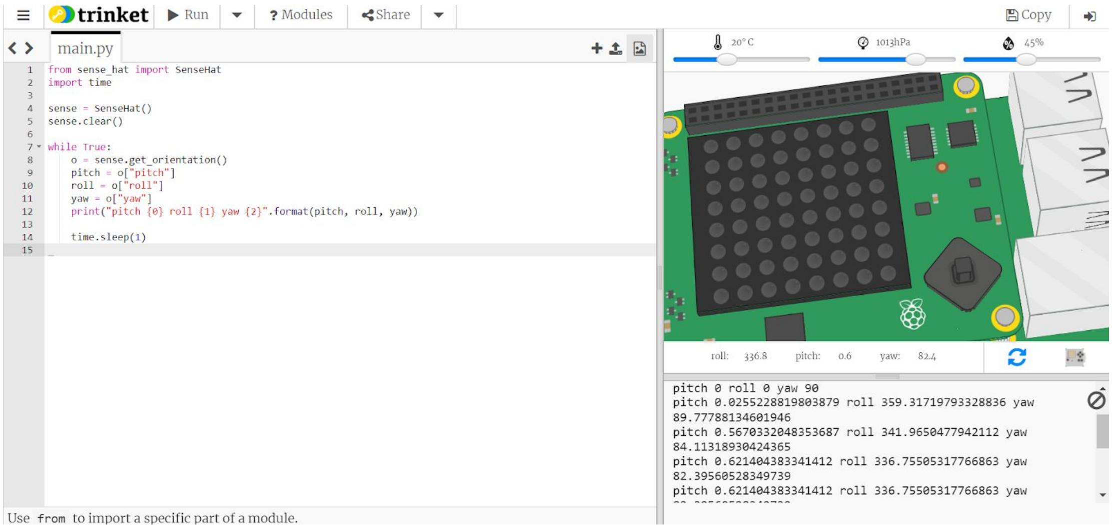
Use `from` to import a specific part of a module.

You can try to use the `set_rotation(angle)` method to rotate the screen by only four ways: **0, 90, 180, or 270** degrees and use the `sense.flip_v()` method to flip the screen **vertically**. This program is to flip the screen **horizontally**.

11. Try to detect pitch, roll, and yaw using IMU sensors. Copy the following program to the `main.py`. Click “**Run**”.

```
from sense_hat import SenseHat
import time

sense = SenseHat()
sense.clear()

while True:
    o = sense.get_orientation()
    pitch = o["pitch"]
    roll = o["roll"]
    yaw = o["yaw"]
    print("pitch {0} roll {1} yaw {2}".format(pitch,
roll, yaw))

    time.sleep(1)
```

You should be able to see the following result:



```
from sense_hat import SenseHat
import time

sense = SenseHat()
sense.clear()

while True:
    o = sense.get_orientation()
    pitch = o["pitch"]
    roll = o["roll"]
    yaw = o["yaw"]
    print("pitch {0} roll {1} yaw {2}".format(pitch, roll, yaw))

    time.sleep(1)
```

roll: 336.8 pitch: 0.6 yaw: 82.4

```
pitch 0 roll 0 yaw 90
pitch 0.0255228819803879 roll 359.31719793328836 yaw
89.77788134601946
pitch 0.5670332048353687 roll 341.9650477942112 yaw
84.11318930424365
pitch 0.621404383341412 roll 336.75505317766863 yaw
82.39560528349739
pitch 0.621404383341412 roll 336.75505317766863 yaw
```

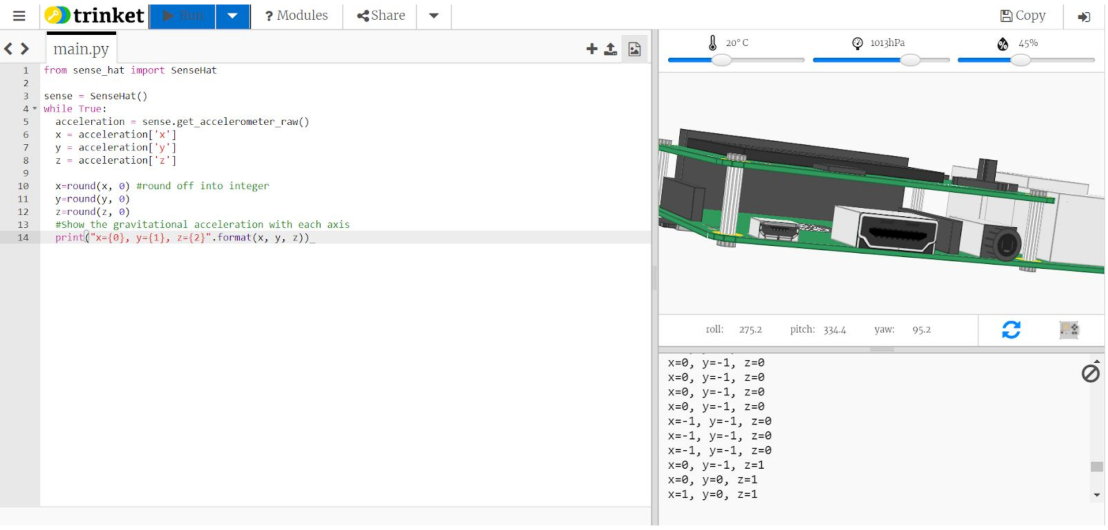
Use `from` to import a specific part of a module.

Noted that the reading of pitch, roll, and yaw change when you rotate the Sense HAT around.

12. Try to detect gravitational acceleration using IMU sensors. Copy the following program to the `main.py`. Click “**Run**”.

```
from sense_hat import SenseHat

sense = SenseHat()
while True:
     acceleration = sense.get_accelerometer_raw()
     x = acceleration['x']
     y = acceleration['y']
     z = acceleration['z']

     x=round(x, 0) #round off into integer
     y=round(y, 0)
     z=round(z, 0)
  #Show the gravitational acceleration with each axis
     print("x={0}, y={1}, z={2}".format(x, y, z))
```

You should be able to see the following result:



```
from sense_hat import SenseHat

sense = SenseHat()
while True:
  acceleration = sense.get_accelerometer_raw()
  x = acceleration['x']
  y = acceleration['y']
  z = acceleration['z']

  x=round(x, 0) #round off into integer
  y=round(y, 0)
  z=round(z, 0)
  #Show the gravitational acceleration with each axis
  print("x={0}, y={1}, z={2}".format(x, y, z))
```

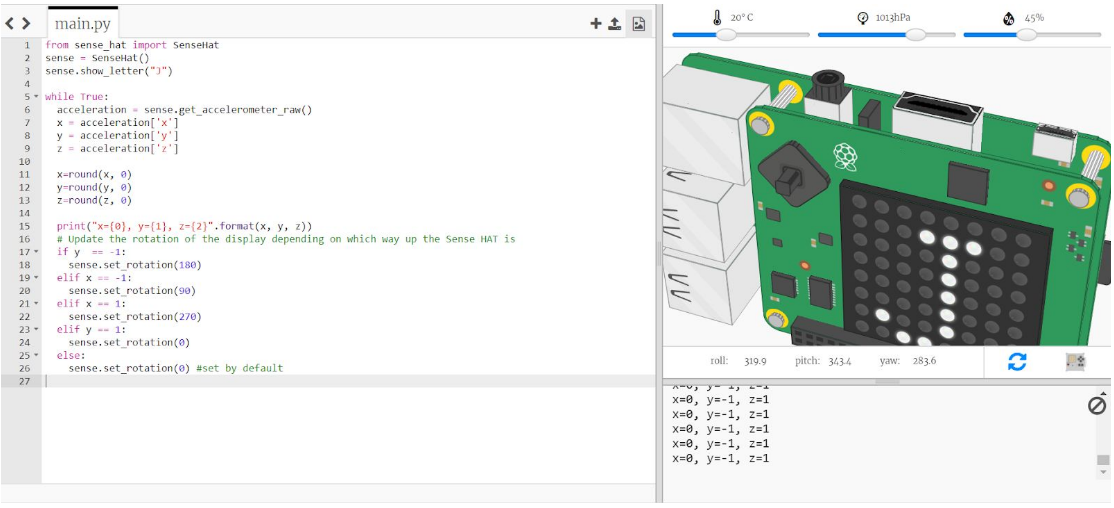
```
x=0, y=-1, z=0
x=0, y=-1, z=0
x=0, y=-1, z=0
x=0, y=-1, z=0
x=-1, y=-1, z=0
x=-1, y=-1, z=0
x=-1, y=-1, z=0
x=0, y=-1, z=1
x=0, y=0, z=1
x=1, y=0, z=1
```

Noted that the reading of x, y, z change when you rotate the Sense HAT around.

13. Try to integrate the LED matrix and the IMU sensor. Copy the following program to the `main.py`. Click “**Run**”.

```
from sense_hat import SenseHat
sense = SenseHat()
sense.show_letter("J")

while True:
     acceleration = sense.get_accelerometer_raw()
     x = acceleration['x']
     y = acceleration['y']
     z = acceleration['z']

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

     print("x={0}, y={1}, z={2}".format(x, y, z))
  # Update the rotation of the display depending on
which way up the Sense HAT is
     if y  == -1:
       sense.set_rotation(180)
     elif x == -1:
       sense.set_rotation(90)
     elif x == 1:
       sense.set_rotation(270)
     elif y == 1:
       sense.set_rotation(0)
     else:
       sense.set_rotation(0) #set by default
```

You should be able to see the following result:



```python
from sense_hat import SenseHat
sense = SenseHat()
sense.show_letter("J")

while True:
  acceleration = sense.get_accelerometer_raw()
  x = acceleration['x']
  y = acceleration['y']
  z = acceleration['z']

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

  print("x={0}, y={1}, z={2}".format(x, y, z))
  # Update the rotation of the display depending on which way up the Sense HAT is
  if y  == -1:
    sense.set_rotation(180)
  elif x == -1:
    sense.set_rotation(90)
  elif x == 1:
    sense.set_rotation(270)
  elif y == 1:
    sense.set_rotation(0)
  else:
    sense.set_rotation(0) #set by default
```

```
x=0, y=-1, z=1
x=0, y=-1, z=1
x=0, y=-1, z=1
x=0, y=-1, z=1
x=0, y=-1, z=1
```

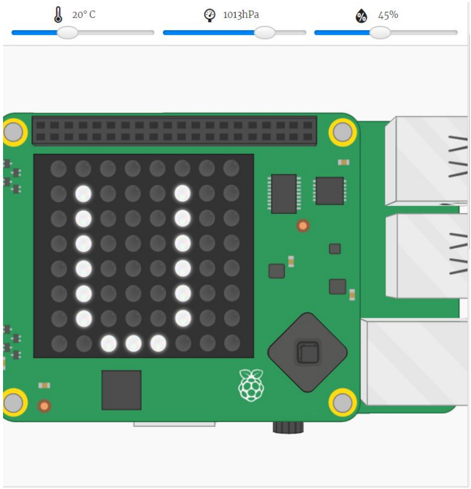
Noted that no matter how you rotate the Sense HAT, you will always see a letter “J” right side up.

14. Try to use the joystick and display the direction on the 8x8 RGB LED Matrix. Copy the following program to the `main.py`. Click “Run”.

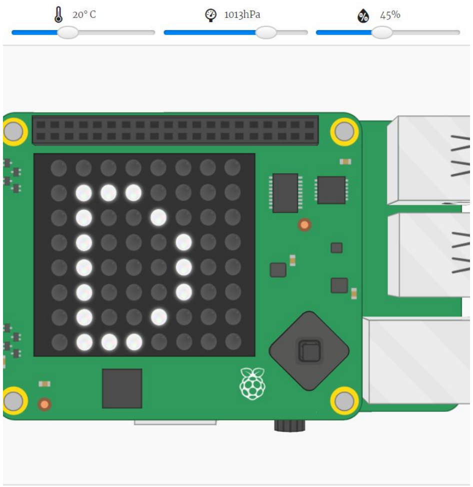
```
from sense_hat import SenseHat
import time

sense = SenseHat()
sense.clear()

while True:
  for event in sense.stick.get_events():
    # Check if the joystick was pressed
    if event.action == "pressed":

      # Check the direction
      if event.direction == "up":
        sense.show_letter("U")      # Up arrow
      elif event.direction == "down":
        sense.show_letter("D")      # Down arrow
      elif event.direction == "left":
        sense.show_letter("L")      # Left arrow
      elif event.direction == "right":
        sense.show_letter("R")      # Right arrow
      elif event.direction == "middle":
        sense.show_letter("M")      # Enter key

      # Wait 1 second and clear the LED matrix
      time.sleep(1)
      sense.clear()
```

Use your mouse to click the Sense HAT once. Then you can use your arrow keys on the keyboard to control it.

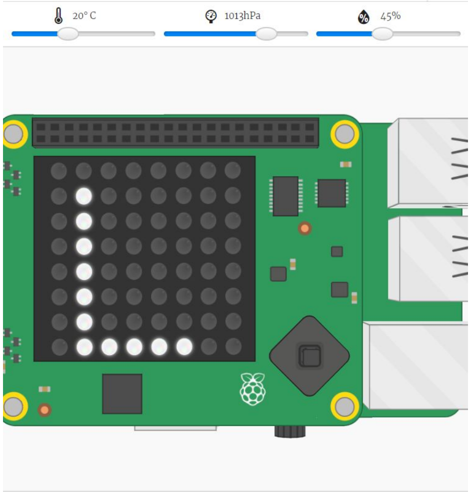
You should be able to see the following result:



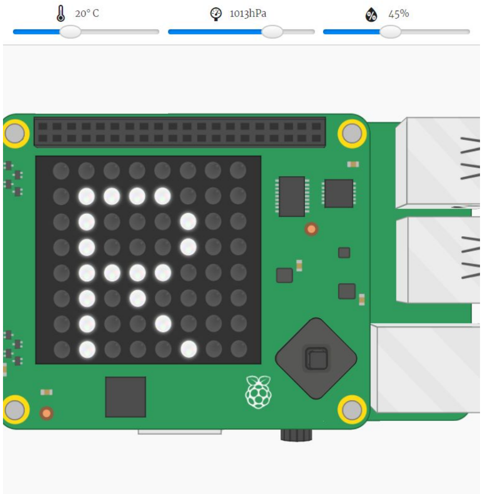
20° C 1013hPa 45%



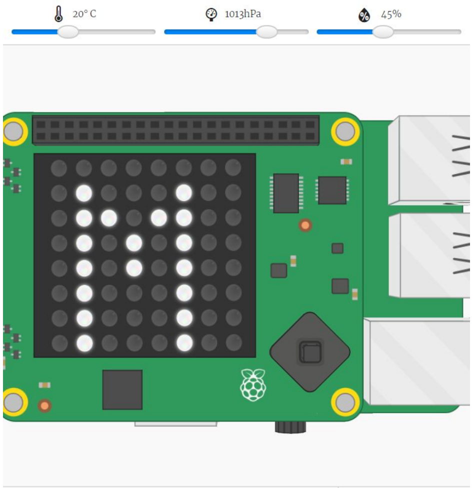
20° C 1013hPa 45%



20° C 1013hPa 45%



20° C 1013hPa 45%



20° C 1013hPa 45%