Sense Hat
Python 3 Cheatsheet
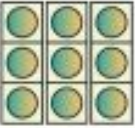
Enabling the
AstroPi mission

To add **Sense HAT** functionality to your python programs add the following lines to import the library for the Sense HAT library:

```python
from sense_hat import SenseHat


sense = SenseHat()
```

From that point forwards you can use any of the set of functions from the Sense HAT Library.

LED Matrix	sense.set_pixel(0, 0, 255, 0, 0)	Sets the top left LED to the colour red.
	sense.show_letter("J", 0, 0, 255)	Displays the letter "J" on the screen in blue.
	sense.show_message("msg", scroll_speed=.1, text_colour=[0, 255, 0])	Displays the message "msg" on the matrix in green.
	sense.load_image("creeper.png", redraw=True)	Load an 8x8 image (in this case "creeper.png") file from the system and display it on the matrix.
	sense.clear()	Clears the LED and switches them all off.
	sense.set_rotation(r=0)	Sets the rotation of the LED matrix.
	sense.set_pixels(pixelList)	Uses the pixel list provided to draw a picture, the pixelList is a list of [R,G,B] values in a list.

Movement	`angles = sense.get_orientation_degrees()`
	`pitch = angles.get("pitch")`
	`roll = angles.get("roll")`
	`yaw = angles.get("yaw")`
		Finds out the current orientation of the Astro Pi board and stores it as a structure called "angles". We can then find out the individual values of pitch, roll and yaw.

	`heading = sense.get_compass() + 180`	FInds out the current compass orientation of the board and returns the angle of north.

Plotting Pixels	Rotating letter "J"

<pre>import time
from sense_hat import SenseHat
 	
sense = SenseHat()
 	
r = (255,0,0)
g = (0,255,0)
b = (0,0,255)
 	
pic = [
r,r,r,r,r,r,r,r,
g,g,g,g,g,g,g,g,
b,b,b,b,b,b,b,b,
r,r,r,r,r,r,r,r,
g,g,g,g,g,g,g,g,
b,b,b,b,b,b,b,b,
r,r,r,r,r,r,r,r,
g,g,g,g,g,g,g,g
]
 	
sense.set_pixels(pic)</pre>
	<pre>import time
	from sense_hat import SenseHat
	
	sense = SenseHat()
	
	purple = (255, 0, 255)
	sense.show_letter("J", purple)
	
	while True:
	    angles = sense.get_orientation_degrees()
	
	    pitch = int(angles.get("pitch"))
	    roll = (angles.get("roll"))
	
	    if 45 <= pitch < 135 and 45 <= roll < 135:
	        sense.set_rotation(r=90)
	        print ("left")
	    elif -135 <= pitch < -45 and 45 <= roll < 135:
	        sense.set_rotation(r=270)
	        print ("right")
	    elif -45 <= pitch < 45 and 45 <= roll < 135:
	        sense.set_rotation(r=0)
	        print ("up")
	    elif -45 <= pitch < 45 and -135 <= roll < -45:
	        sense.set_rotation(r=180)
	        print ("down")
	
	    time.sleep(0.1)</pre>