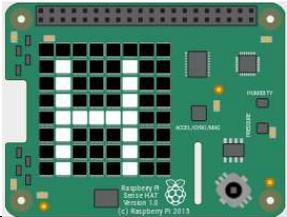
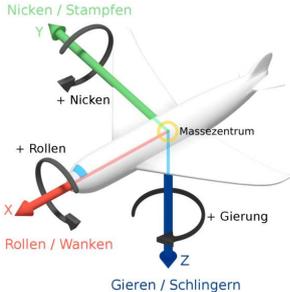


Sense HAT (Sense HAT-Emulator) am Raspberry Pi	
Entwicklungsumgebung auf LINUX	Start/ Entwicklung/ Thonny Python DIE Emulation unter: https://trinket.io/sense-hat
Bibliotheken	
Sense_HAT	<code>from sense_hat import senseHat</code>
Sense_HAT Emulator	<code>from sense_emu import SenseHat</code> # Emulator unter Start/ Entwicklung/ Sense ..
Zeit	<code>import time</code> # z.B. <code>from time import sleep</code>
GPIO	<code>import gpio</code> # z.B. <code>LED from gpio import LED, Button, Buzzer</code>
Zufall	<code>import random</code> # z.B. <code>uniform (sleep(uniform(5, 10))</code> Zeit zw. 5-10 Sek
OS	<code>import os</code> # z.B. <code>_exit</code> (am Ende einer Funktion -> <code>-exit(0)</code>)
Ausgabe	
Textausgabe 	<pre> from sense_hat import SenseHat # Bibliothek laden sense = SenseHat() # Objekt sense erzeugen sense.show_message(" Hallo ") # Ausgabe als Scrolltext sense.show_message(" Hallo ", text_colour=(255, 255, 0)) # Textfarbe ändern sense.show_message(" Hallo ", back_colour=(0, 0, 255) # Hintergrundfarbe # Farbvariable -> gelb = (255, 255, 0) -> text_colour(gelb) sense.show_message(" Hallo ", scroll_speed=(0.05)) #Scroll_Geschwindigkeit </pre>
Buchstabenausgabe	<pre> from sense_hat import SenseHat # Bibliothek laden sense = SenseHat() # Objekt sense erzeugen sense.show_letter(" Z ") # Buchstabe scrollt nicht; Farbmöglichkeiten wie Text </pre>
Bildschirm löschen	<pre> from sense_hat import SenseHat # Bibliothek laden sense = SenseHat() # Objekt sense erzeugen sense.clear() # ohne Parameter löschen sleep(1) # Pause sense.clear(0,0,255) # ganzer Bildschirm blau gefärbt </pre>
Helligkeit ändern Schwachlichtmodus	<pre> from sense_hat import SenseHat # Bibliothek laden sense = SenseHat() # Objekt sense erzeugen sense.clear(255,255,255) # alle Pixel weiss sense.low_light = True # Helligkeit absenken </pre>
Bildpunkte ansprechen X 0 1 2 3 4 5 6 7 Y 0 1 2 3 4 5 6 7	<pre> from sense_hat import SenseHat # Bibliothek laden sense = SenseHat() # Objekt sense erzeugen sense.clear() # löscht alle LEDs sense.clear(0, 255, 0) # es werden aber auch Farbparameter akzeptiert sense.set_pixel(0, 2, (0, 0, 255)) # X-Pos, Y-Pos, (RGB-Wert) für ein Pixel #CodeBlock für alle 64 Pixel auf der Matrix g = (0, 255, 0) # Variable für Grün b = (0, 0, 0) # Variable für Schwarz bild1 = [g, g, g, g, g, g, g, g, g, g, g, g, g, g, g, g, g, g, g, g, g, b, b, g, g, g, g, g, g, b, b, g, g, g, g, b, b, g, g, g, g, g, b, b, b, b, g, g, g, g, b, b, b, b, g, g, g, g, b, g, g, b, g, g,] sense.set_pixels(bild1) #Animation from time import sleep # ganz oben zusätzliche Bibliothek einbinden while True: # am Ende dies 3 Zeilen sleep(1) #Pause 1 sek sense.flip_h() #sense.flip_h spiegelt auf horizontaler Achse # sense.flip_v spiegelt auf vertikaler Achse; sense.rotation(45) dreht das Bild </pre>

Sensoren		
Drucksensor	<pre>from sense_hat import SenseHat sense = SenseHat() print(sense.get_pressure())</pre>	<pre># Bibliothek laden # Objekt sense erzeugen # Ausgabe des Drucks auf Konsole</pre>
Luftfeuchtigkeit	<pre>from sense_hat import SenseHat sense = SenseHat() print (sense.get_humidity())</pre>	<pre># Bibliothek laden # Objekt sense erzeugen # Ausgabe d. Luftfeuchtigkeit a. Konsole</pre>
Temperatur	<pre>from sense_hat import SenseHat sense = SenseHat() print (sense.get_temperature ())</pre>	<pre># Bibliothek laden # Objekt sense erzeugen # Ausgabe der Temperatur auf Konsole</pre>
Bewertungsauswertung		
<p>Rotation</p> 	<pre>from sense_hat import SenseHat sense = SenseHat() while True: pitch, roll, yaw = sense.get_orientation().values() # Werteübergabe print(" P= ",round(pitch,1), " R= ",round(roll,1), " Y= ",round(yaw,1))# Ausgabe sleep(1) # Pause</pre>	
Beschleunigung	<pre>from sense_hat import SenseHat sense = SenseHat() while True: acceleration = sense.get_accelerometer_raw() # Werteübergabe x = acceleration[" x "] # Auswertung X-Achse aus Liste y = acceleration[" y "] # Auswertung Y-Achse aus Liste z = acceleration[" z "] # Auswertung Z-Achse aus Liste print(" X= ",round(x,2), " Y= ",round(y,2), " Z= ",round(z,2)) # Ausgabe sleep(1) # Pause</pre>	
Eingabe		
Joystick	<pre>from sense_hat import SenseHat sense = SenseHat() while True: for event in sense.stick.get_events(): #Anweisung auf Joystick zu warten print(event.direction) # Ausgabe d. Richtung -up,down-left,right,middle print(event.action) #Ausgabe Aktion-gedrückt, losgelassen, gehalten # Beispiel: from sense_hat import SenseHat sense = SenseHat() def rot(): # anlegen Funktion rot sense.clear(255, 0, 0) # definieren Löschfunktion alle Pix rot def blau(): # anlegen Funktion blau sense.clear(0, 0, 255) # definieren Löschfunktion alle Pix blau def aus(): # anlegen Funktion aus sense.clear(255, 255, 255) # definieren Löschfunktion alle Pix aus sense.stick.direction_up = rot # wenn up Funktion - rot sense.stick.direction_down = blau # wenn down Funktion - blau sense.stick.direction_middle = aus # wenn Feuertaste Funktion - aus</pre>	