Raspberry Pi Integration Code used in Thonny for controlling light and fan through website

```
#!/usr/bin/env python3
import RPi.GPIO as GPIO
import os
from http.server import BaseHTTPRequestHandler, HTTPServer
host_name = '192.168.104.111'  # IP Address of Raspberry Pi
host_port = 8000
# Define GPIO pins for LEDs
LED_PIN_1 = 18  # Assuming GPIO pin 23 for the first LED
LED_PIN_2 = 23  # Assuming GPIO pin 23 for the second LED
# Define GPIO pins for Motor Driver L298N
MOTOR_PIN_1 = 4 # Motor input pin 1
MOTOR_PIN_2 = 17 # Motor input pin 2
MOTOR_ENABLE_PIN = 22 # Optional: Motor enable pin, if used
def setupGPIO():
 GPIO.setmode(GPIO.BCM)
 GPIO.setwarnings(False)

 # Setup LEDs
 GPIO.setup(LED_PIN_1, GPIO.OUT)
 GPIO.setup(LED_PIN_2, GPIO.OUT)

 # Setup Motor Driver
 GPIO.setup(MOTOR_PIN_1, GPIO.OUT)
 GPIO.setup(MOTOR_PIN_2, GPIO.OUT)
 GPIO.setup(MOTOR_ENABLE_PIN, GPIO.OUT)

 # Initialize motor state
 GPIO.output(MOTOR_PIN_1, GPIO.LOW)
 GPIO.output(MOTOR_PIN_2, GPIO.LOW)
 GPIO.output(MOTOR_ENABLE_PIN, GPIO.HIGH) # Enable the motor driver (if required)
def getTemperature():
 temp = os.popen("/opt/vc/bin/vcgencmd measure_temp").read()
```

```
 return temp
class MyServer(BaseHTTPRequestHandler):
 def do_HEAD(self):
 self.send_response(200)
 self.send_header('Content-type', 'text/html')
 self.end_headers()
 def _redirect(self, path):
 self.send_response(303)
 self.send_header('Content-type', 'text/html')
 self.send_header('Location', path)
 self.end_headers()
 def do_GET(self):
 html = '''
 <html>
 <body style="width:960px; margin: 20px auto;">
 <h1>Welcome to my Room</h1>
 <p>{}</p>
 <form action="/" method="POST">
 <h2>Light Switch 1:</h2>
 <input type="submit" name="led1" value="On">
 <input type="submit" name="led1" value="Off">
 <h2>Light Switch 2:</h2>
 <input type="submit" name="led2" value="On">
 <input type="submit" name="led2" value="Off">
 <h2>Fan Switch:</h2>
 <input type="submit" name="motor" value="On">
 <input type="submit" name="motor" value="Off">
 </form>
 </body>
 </html>
 '''
 temp = getTemperature()
 self.do_HEAD()
```

```
self.wfile.write(html.format(temp[5:]).encode("utf-8"))
def do_POST(self):
content_length = int(self.headers['Content-Length'])
post_data = self.rfile.read(content_length).decode("utf-8")

# Parse POST data
params = dict(param.split('=') for param in post_data.split('&'))

setupGPIO()
# Control LED 1
if 'led1' in params:
if params['led1'] == 'On':
GPIO.output(LED_PIN_1, GPIO.HIGH)
elif params['led1'] == 'Off':
GPIO.output(LED_PIN_1, GPIO.LOW)

# Control LED 2
if 'led2' in params:
if params['led2'] == 'On':
GPIO.output(LED_PIN_2, GPIO.HIGH)
elif params['led2'] == 'Off':
GPIO.output(LED_PIN_2, GPIO.LOW)

# Control Motor
if 'motor' in params:
if params['motor'] == 'On':
GPIO.output(MOTOR_PIN_1, GPIO.HIGH)
GPIO.output(MOTOR_PIN_2, GPIO.LOW)
elif params['motor'] == 'Off':
GPIO.output(MOTOR_PIN_1, GPIO.LOW)
GPIO.output(MOTOR_PIN_2, GPIO.LOW)

print("LED 1 is {}".format('On' if GPIO.input(LED_PIN_1) else 'Off'))
```

```
 print("LED 2 is {}".format('On' if GPIO.input(LED_PIN_2) else 'Off'))
 print("Motor is {}".format('On' if GPIO.input(MOTOR_PIN_1) else 'Off'))

 self._redirect('/') # Redirect back to the root URL
# # # # # Main # # # # #
if __name__ == '__main__':
 http_server = HTTPServer((host_name, host_port), MyServer)
 print("Server Starts - %s:%s" % (host_name, host_port))
 try:
 http_server.serve_forever()
 except KeyboardInterrupt:
 http_server.server_close()
 GPIO.cleanup()
```