CODE:

import time

from machine import Pin

import dht

# Initialize the DHT22 sensor

dht\_sensor = dht.DHT22(Pin(2))

# Initialize the LEDs

led\_red = Pin(14, Pin.OUT)

led\_green = Pin(15, Pin.OUT)

while True:

    try:

        # Read temperature and humidity from the sensor

        dht\_sensor.measure()

        temperature\_celsius = dht\_sensor.temperature()

        humidity\_percent = dht\_sensor.humidity()

        # Print the values

        print("Temperature: {}°C, Humidity: {}%".format(temperature\_celsius, humidity\_percent))

        # Control LEDs based on temperature and humidity

        if temperature\_celsius > 25:

            led\_red.on()

            led\_green.off()

        else:

            led\_red.off()

            led\_green.on()

    except Exception as e:

        print("Error:", e)

    # Delay for 2 seconds before reading again

    time.sleep(2)

LOGIC :

1. **Import Libraries**:
   * time: Library to work with time-related functions.
   * machine.Pin: Library to control GPIO pins on the microcontroller.
   * dht: Library to interface with the DHT22 sensor.
2. **Initialize Sensor and LEDs**:
   * dht\_sensor = dht.DHT22(Pin(2)): Initializes the DHT22 sensor connected to GPIO pin 2.
   * led\_red = Pin(14, Pin.OUT): Initializes the red LED connected to GPIO pin 14 as an output pin.
   * led\_green = Pin(15, Pin.OUT): Initializes the green LED connected to GPIO pin 15 as an output pin.
3. **Main Loop**:
   * while True:: This creates an infinite loop, ensuring the code runs continuously.
4. **Reading Sensor Data**:
   * dht\_sensor.measure(): Measures temperature and humidity from the DHT22 sensor.
   * temperature\_celsius = dht\_sensor.temperature(): Reads the temperature in Celsius.
   * humidity\_percent = dht\_sensor.humidity(): Reads the humidity in percentage.
5. **Printing Data**:
   * print("Temperature: {}°C, Humidity: {}%".format(temperature\_celsius, humidity\_percent)): Prints the temperature and humidity values.
6. **Controlling LEDs**:
   * if temperature\_celsius > 25:: Checks if the temperature is greater than 25°C.
     + led\_red.on(): Turns on the red LED if the temperature is above 25°C.
     + led\_green.off(): Turns off the green LED if the temperature is above 25°C.
   * else:: If the temperature is not above 25°C:
     + led\_red.off(): Turns off the red LED.
     + led\_green.on(): Turns on the green LED.
7. **Error Handling**:
   * except Exception as e:: Catches any exceptions that might occur during execution and prints the error message.
8. **Delay**:
   * time.sleep(2): Pauses the execution for 2 seconds before the next iteration of the loop.

This loop continuously reads the temperature and humidity, prints the values, controls the LEDs based on the temperature, and then waits for 2 seconds before repeating the process.

Top of Form

Bottom of Form

IMAGE: