

This program is used with the Auduino board.

```
import time
import random
from pyfirmata import Arduino, util

// TEMPERATURE VARIABLE
class TemperatureSensor:
    def __init__(self):
        self.current_temperature = 0

    def get_temperature(self):
        # Replace this with actual sensor readings
        self.current_temperature = random.uniform(15, 35)
        return self.current_temperature

// LIGHT VARIABLE
class LightSensor:
    def __init__(self):
        self.current_light_intensity = 0

    def get_light_intensity(self):
        # Replace this with actual sensor readings
        self.current_light_intensity = random.uniform(100, 1000)
        return self.current_light_intensity

class GPIOController:
    def __init__(self, board):
        self.window_status = "closed"
        self.growing_lights_status = "off"
        self.current_temperature = 0
        self.current_light_intensity = 0
        self.board = board

        # Define LED pins
        self.temperature_led_pin = 13 # Replace with your actual pin number
        self.light_led_pin = 12 # Replace with your actual pin number

    def open_window(self):
        print("Opening the window")
        self.window_status = "open"

    def close_window(self):
        print("Closing the window")
        self.window_status = "closed"

    def turn_on_lights(self):
        print("Turning on growing lights")
        self.growing_lights_status = "on"

    def turn_off_lights(self):
        print("Turning off growing lights")
        self.growing_lights_status = "off"
```

```
def display_temperature_on_led_grid(self, temperature):
    self.current_temperature = temperature
    self.board.digital[self.temperature_led_pin].write(1) # Turn on LED
    time.sleep(1)
    self.board.digital[self.temperature_led_pin].write(0) # Turn off LED
    print(f"Temperature: {temperature} displayed on LED grid")

def display_light_on_led_grid(self, light_intensity):
    self.current_light_intensity = light_intensity
    self.board.digital[self.light_led_pin].write(1) # Turn on LED
    time.sleep(1)
    self.board.digital[self.light_led_pin].write(0) # Turn off LED
    print(f"Light Intensity: {light_intensity} displayed on LED grid")

def main():
    # Connect to Arduino board
    board = Arduino('COM3') # Replace with your actual port

    # Setup the board
    it = util.Iterator(board)
    it.start()

    temperature_sensor = TemperatureSensor()
    light_sensor = LightSensor()
    gpio_controller = GPIOController(board)

    while True:
        # Read temperature and light intensity values from sensors
        temperature = temperature_sensor.get_temperature()
        light_intensity = light_sensor.get_light_intensity()

        # Display values on LED grid and store in variables
        gpio_controller.display_temperature_on_led_grid(temperature)
        gpio_controller.display_light_on_led_grid(light_intensity)

        # Temperature-based actions
        if temperature < 20:
            gpio_controller.close_window()
        elif temperature > 30:
            gpio_controller.open_window()

        # Light intensity-based actions
        if light_intensity > 700:
            gpio_controller.turn_off_lights()
        elif light_intensity < 200:
            gpio_controller.turn_on_lights()

        # Simulating a delay between sensor readings
        time.sleep(5)

if __name__ == "__main__":
    main()
```

