const int pirPin = 2; // PIR sensor output pin

const int ledPin = 9; // LED pin

const int buzPin = 3;

int motionState = LOW; // Initial state of motion

void setup() {

Serial.begin(9600); // Start serial communication

pinMode(pirPin, INPUT); // Set PIR sensor pin as input

pinMode(ledPin, OUTPUT); // Set LED pin as output

pinMode(buzPin, OUTPUT);

}

void loop() {

motionState = digitalRead(pirPin); // Read PIR sensor state

if (motionState == HIGH) {

digitalWrite(ledPin, HIGH); // Turn on LED

digitalWrite(buzPin, HIGH);

Serial.println("Motion detected!"); // Print message

delay(100); // Delay to prevent flooding Serial Monitor

} else {

digitalWrite(ledPin, LOW); // Turn off LED

digitalWrite(buzPin, LOW);

}

delay(100); // Small delay to improve responsiveness

}

//2nd code

int pirPin = 2; // PIR sensor output pin

int ledPin = 13; // LED pin

int buzzerPin = 12; // Buzzer pin

void setup() {

pinMode(pirPin, INPUT); // Set PIR sensor pin as input

pinMode(ledPin, OUTPUT); // Set LED pin as output

pinMode(buzzerPin, OUTPUT); // Set buzzer pin as output

Serial.begin(9600); // Initialize serial communication for debugging

}

void loop() {

int pirState = digitalRead(pirPin); // Read PIR sensor state

if (pirState == HIGH) {

digitalWrite(ledPin, HIGH); // Turn on LED

digitalWrite(buzzerPin, HIGH); // Turn on buzzer

Serial.println("Motion detected!"); // Print message to serial monitor

} else {

digitalWrite(ledPin, LOW); // Turn off LED

digitalWrite(buzzerPin, LOW); // Turn off buzzer

Serial.println("No motion."); // Print message to serial monitor

}

delay(1000); // Wait for a second before checking again

}