

PROJECT CODE

```
// Define LDR analog sensor pins

const int ldrLeftPin = A0; // Left LDR connected to analog pin A0

const int ldrRightPin = A1; // Right LDR connected to analog pin A1

// Define motor driver pins

const int motorLeftPin = 2; // Motor control pin 1

const int motorRightPin = 3; // Motor control pin 2

void setup() {

    // Set motor pins as output

    pinMode(motorLeftPin, OUTPUT);

    pinMode(motorRightPin, OUTPUT);

    // Start communication with the serial monitor (for debugging)

    Serial.begin(9600);

}

void loop() {

    // Read light levels from LDRs

    int leftLight = analogRead(ldrLeftPin); // Read left LDR value

    int rightLight = analogRead(ldrRightPin); // Read right LDR value

    // Print the LDR values for debugging

    Serial.print("Left LDR: ");

    Serial.print(leftLight);

    Serial.print(" | Right LDR: ");

    Serial.println(rightLight);

    // Compare the LDR values to decide motor movement
```

```
    if (leftLight > rightLight) {  
        moveLeft();  
    }  
    else if (rightLight > leftLight) {  
        moveRight();  
    }  
    else {  
        stopMotor();  
    }  
    delay(100); // Wait for a short period to stabilize the response  
}  
  
// Function to move the motor to the left  
void moveLeft() {  
    digitalWrite(motorLeftPin, HIGH);  
    digitalWrite(motorRightPin, LOW);  
    Serial.println("Moving Left");  
}  
  
// Function to move the motor to the right  
void moveRight() {  
    digitalWrite(motorLeftPin, LOW);  
    digitalWrite(motorRightPin, HIGH);  
    Serial.println("Moving Right");  
}  
  
// Function to stop the motor
```

```
void stopMotor() {  
    digitalWrite(motorLeftPin, LOW);  
    digitalWrite(motorRightPin, LOW);  
    Serial.println("Motor Stopped");  
}
```