```
/*
 * ESP8266 Hydroponic Automation Sensor Module
 * Author: Varad Chaskar
 * Description: This code is part of a hydroponic automation project and is designed for the ESP8266 microcontroller.
 *              It gathers environmental data using sensors, including temperature, humidity, light intensity, and water level.
 *              The collected data is transmitted to the ThingSpeak cloud platform for real-time monitoring and analysis.
 * Date: January 28, 2024
 */

#include <WiFiManager.h>                                // Library for simplified WiFi connection handling
#include <ESP8266WiFi.h>                                // ESP8266 WiFi library
#include <DHT.h>                                        // DHT sensor library for interfacing with DHT11 sensor
#include "ThingSpeak.h"                                 // Library for seamless communication with ThingSpeak IoT platform

WiFiClient client;                                      // WiFi client for ThingSpeak communication

unsigned long myChannelNumber = 2384399;                        // Channel number on ThingSpeak cloud
const char *myWriteAPIKey = "ELW2NF5Q83OGB39G";                 // Write API key for ThingSpeak
const char *myCounterReadAPIKey = "3D8NH4JCI0EDYMIU";           // Read API key for ThingSpeak

DHT dht;                                                // Initialize DHT sensor
const int DHTPIN = D5;                                  // Pin for DHT sensor

void setup() {
  pinMode(LED_BUILTIN, OUTPUT);                         // Set built-in LED pin as output

  Serial.begin(115200);                                 // Initialize serial communication for debugging
  WiFiManager wm;                                       // Create WiFiManager instance for easy WiFi connection handling
  bool res;
  res = wm.autoConnect("ASAA");                         // Attempt to connect to WiFi using saved credentials
  if(!res) {
    Serial.println("Failed to connect");
    digitalWrite(LED_BUILTIN, HIGH);                    // Turn on built-in LED to indicate connection failure
    delay(1000);
    digitalWrite(LED_BUILTIN, LOW);                     // Turn off built-in LED
    delay(1000);
    digitalWrite(LED_BUILTIN, HIGH);                    // Turn on built-in LED
    ESP.restart();                                      // Restart the ESP8266
  }
  else {
    ThingSpeak.begin(client);                           // Initialize ThingSpeak communication
    digitalWrite(LED_BUILTIN, LOW);                     // Turn off built-in LED to indicate successful connection
    Serial.println("Connected to WiFi");
    dht.setup(DHTPIN);                                  // Initialize DHT sensor
  }
}

void loop() {
  digitalWrite(LED_BUILTIN, LOW);                       // Turn off built-in LED

  // Read DHT sensor data
  float temperature = dht.getTemperature();             // Read temperature from DHT sensor
  float humidity = dht.getHumidity();                   // Read humidity from DHT sensor
```

```
  // Read LDR analog input
  int ldrValue = analogRead(A0);                    // Read analog input from LDR

  // Read water level (0 or 1) from ThingSpeak
  int waterLevel = ThingSpeak.readLongField(myChannelNumber, 4, myCounterReadAPIKey);

  // Send data to ThingSpeak
  ThingSpeak.writeField(myChannelNumber, 1, temperature, myWriteAPIKey);      // Send temperature data to ThingSpeak
  ThingSpeak.writeField(myChannelNumber, 2, humidity, myWriteAPIKey);         // Send humidity data to ThingSpeak
  ThingSpeak.writeField(myChannelNumber, 3, ldrValue, myWriteAPIKey);         // Send LDR value to ThingSpeak
  ThingSpeak.writeField(myChannelNumber, 4, waterLevel, myWriteAPIKey);       // Send water level data to ThingSpeak
}
```