```
// Include necessary libraries
#include <WiFi.h>         // For connecting ESP32 to WiFi
#include <HTTPClient.h>   // For sending HTTP requests
#include <DHT11.h>        // Library for interfacing with DHT11 sensor

// Initialize the DHT11 sensor on GPIO pin 18
DHT11 dht11(18);

// Wi-Fi credentials
const char* wifiname = "Aaradhya's iPhone";
const char* password = "12345678";
// ThingSpeak API server
const char* serverName = "http://api.thingspeak.com/update"; // URL for sending data to
ThingSpeak

// Your ThingSpeak Write API Key
String apiKey = "9W9BF9N9W1VCUD7Q"; // Replace with your ThingSpeak Write API Key

void setup() {
  Serial.begin(115200);              // Start serial communication at 115200 baud
  WiFi.begin(wifiname, password);    // Begin Wi-Fi connection

  Serial.println("Connecting");

  // Wait until WiFi is connected
  while(WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  // Print local IP once connected
  Serial.println("");
  Serial.print("Connected to WiFi network with IP Address: ");
  Serial.println(WiFi.localIP());
}

void loop() {
  // Check if WiFi is still connected
  if(WiFi.status() == WL_CONNECTED){
      WiFiClient client;        // Create a WiFi client object
      HTTPClient http;          // Create HTTP client for sending requests
      delay(1000);              // Wait for 1 second between readings

      int temperature = 0;      // Variable to store temperature
      int humidity = 0;         // Variable to store humidity
      // Read temperature and humidity from DHT11 sensor
      int result = dht11.readTemperatureHumidity(temperature, humidity);
      if (result == 0) {
```

```
      // If sensor read was successful, print the values
      Serial.print("Temperature: ");
      Serial.print(temperature);
      Serial.print(" Â°C\tHumidity: ");
      Serial.print(humidity);
      Serial.println(" %");
      delay(2000); // Optional: add small delay before sending to server
    } else {
      // If read failed, print error message
      Serial.println(DHT11::getErrorString(result));
    }
    // Begin HTTP POST request
    http.begin(client, serverName);

    // Set content-type header
    http.addHeader("Content-Type", "application/x-www-form-urlencoded");

    // Format data to be sent to ThingSpeak (sending only temperature here)
    String httpRequestData = "api_key=" + apiKey + "&field1=" + String(temperature);

    // Send the POST request and store response code
    int httpResponseCode = http.POST(httpRequestData);

    // Print the response code (200 = OK)
    Serial.print("HTTP Response code: ");
    Serial.println(httpResponseCode);

    // End the HTTP connection
    http.end();
  }
  else {
    // If not connected to WiFi, print message
    Serial.println("WiFi Disconnected");
  }
}
```