|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | |  |  |  | https://mail.google.com/mail/u/0/images/cleardot.gif  https://mail.google.com/mail/u/0/images/cleardot.gif | | |  | | --- | | to me  https://mail.google.com/mail/u/0/images/cleardot.gif | | | |   #include <SoftwareSerial.h>       //Software Serial library SoftwareSerial espSerial(2, 3);   //Pin 2 and 3 act as RX and TX. Connect them to TX and RX of ESP8266       #define DEBUG true String mySSID = "DESKTOP123";       // WiFi SSID String myPWD = "vv123456"; // WiFi Password String myAPI = "B2VXKAG8EVBEHB4V";   // API Key String myHOST = "[api.thingspeak.com](http://api.thingspeak.com/)"; String myPORT = "80"; String myFIELD = "field1";  int sendVal;   void setup() {   Serial.begin(9600);   espSerial.begin(115200);      espData("AT+RST", 1000, DEBUG);                      //Reset the ESP8266 module   espData("AT+CWMODE=1", 1000, DEBUG);                 //Set the ESP mode as station mode   espData("AT+CWJAP=\""+ mySSID +"\",\""+ myPWD +"\"", 1000, DEBUG);   //Connect to WiFi network   /\*while(!esp.find("OK"))    {                 //Wait for connection   }\*/   delay(1000);    }    void loop()   {     /\* Here, I'm using the function random(range) to send a random value to the       ThingSpeak API. You can change this value to any sensor data      so that the API will show the sensor data       \*/          sendVal = random(1000); // Send a random number between 1 and 1000     String sendData = "GET /update?api\_key="+ myAPI +"&"+ myFIELD +"="+String(sendVal);     espData("AT+CIPMUX=1", 1000, DEBUG);       //Allow multiple connections     delay(3000);     espData("AT+CIPSTART=,\"TCP\",\""+ myHOST +"\","+ myPORT, 1000, DEBUG);     delay(3000);     espData("AT+CIPSEND=" +String(sendData.length()+4),3000,DEBUG);       delay(3000);     espSerial.find(">");      espSerial.println(sendData);     Serial.print("Value to be sent: ");     Serial.println(sendVal);     Serial.println(sendData);           espData("AT+CIPCLOSE",1000,DEBUG);     delay(10000);   }    String espData(String command, const int timeout, boolean debug) {   Serial.print("AT Command ==> ");   Serial.print(command);   Serial.println("     ");      String response = "";   espSerial.println(command+"\r\n");   long int time = millis();   while ( (time + timeout) > millis())   {     while (espSerial.available())     {       char c = espSerial.read();       response += c;     }   }   if (debug)   {     Serial.print(response);   }   return response; }       |  |  | | --- | --- | |  |  | |  |  |