變數不能大寫。

有單引號(')、換行(\n)要設為脫逸字元。

~~字串變數不用留雙引號~~

★字首單引號前加空格( ),避免被excel去掉。

此空格

若積木只有單行程式碼,換行依據：尾部有沒有；分號

變數若是#define BUTTON\_PIN 3，記得在變數名稱前加空格( ) '#define BUTTON\_PIN '+value\_pin

若每行程式碼前是 → Tab鍵，要改成空格( )

+'\

|  |  |
| --- | --- |
| \_01imi\_mqttcloud\_init  iMi雲端平台ESP32 MQTT板初始化  UART通訊  WiFi設定  MQTT設定  \_02imi\_mqttcloud\_uart  RX: TX:  **16**  **17**  '//UART通訊'  'SoftwareSerial Esp32GpsSerial('+value\_rx+','+value\_tx+');'  \_03imi\_mqttcloud\_wifi  ssid: 密碼:  “ “  “ “  '//WiFi設定'  'const char\* ssid = '+value\_ssid+';'  'const char\* password = '+value\_wifipwd+';'  \_04imi\_mqttcloud\_mqtt  帳號 密碼 port 服務器  “ “  “ “  8883  “ “  '//HiveMQ設定'  'const char\* mqtt\_server =' +value\_mqtt\_server+';'  'const char\* mqtt\_username =' +value\_mqtt\_user+';'  'const char\* mqtt\_password =' +value\_mqtt\_pwd+';'  'const int mqtt\_port =' +value\_port+';' | #include <WiFi.h>  #include <PubSubClient.h>  #include <WiFiClientSecure.h>  #include <SoftwareSerial.h>  //=====變數設定 Start=====  Statements區塊  //=====變數設定 End=====  const char\* TOPIC\_CAR\_GPS = "imiRobot/car/gps";  //車子的格子座標xy(例:35)  int carX=0;  int carY=0;  char msgXY[3];  WiFiClientSecure espClient;  PubSubClient client(espClient);  //HiveMQ Cloud  static const char\* root\_ca PROGMEM = R"EOF(  -----BEGIN CERTIFICATE-----  MIIFazCCA1OgAwIBAgIRAIIQz7DSQONZRGPgu2OCiwAwDQYJKoZIhvcNAQELBQAw  TzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh  cmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwHhcNMTUwNjA0MTEwNDM4  WhcNMzUwNjA0MTEwNDM4WjBPMQswCQYDVQQGEwJVUzEpMCcGA1UEChMgSW50ZXJu  ZXQgU2VjdXJpdHkgUmVzZWFyY2ggR3JvdXAxFTATBgNVBAMTDElTUkcgUm9vdCBY  MTCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBAK3oJHP0FDfzm54rVygc  h77ct984kIxuPOZXoHj3dcKi/vVqbvYATyjb3miGbESTtrFj/RQSa78f0uoxmyF+  0TM8ukj13Xnfs7j/EvEhmkvBioZxaUpmZmyPfjxwv60pIgbz5MDmgK7iS4+3mX6U  A5/TR5d8mUgjU+g4rk8Kb4Mu0UlXjIB0ttov0DiNewNwIRt18jA8+o+u3dpjq+sW  T8KOEUt+zwvo/7V3LvSye0rgTBIlDHCNAymg4VMk7BPZ7hm/ELNKjD+Jo2FR3qyH  B5T0Y3HsLuJvW5iB4YlcNHlsdu87kGJ55tukmi8mxdAQ4Q7e2RCOFvu396j3x+UC  B5iPNgiV5+I3lg02dZ77DnKxHZu8A/lJBdiB3QW0KtZB6awBdpUKD9jf1b0SHzUv  KBds0pjBqAlkd25HN7rOrFleaJ1/ctaJxQZBKT5ZPt0m9STJEadao0xAH0ahmbWn  OlFuhjuefXKnEgV4We0+UXgVCwOPjdAvBbI+e0ocS3MFEvzG6uBQE3xDk3SzynTn  jh8BCNAw1FtxNrQHusEwMFxIt4I7mKZ9YIqioymCzLq9gwQbooMDQaHWBfEbwrbw  qHyGO0aoSCqI3Haadr8faqU9GY/rOPNk3sgrDQoo//fb4hVC1CLQJ13hef4Y53CI  rU7m2Ys6xt0nUW7/vGT1M0NPAgMBAAGjQjBAMA4GA1UdDwEB/wQEAwIBBjAPBgNV  HRMBAf8EBTADAQH/MB0GA1UdDgQWBBR5tFnme7bl5AFzgAiIyBpY9umbbjANBgkq  hkiG9w0BAQsFAAOCAgEAVR9YqbyyqFDQDLHYGmkgJykIrGF1XIpu+ILlaS/V9lZL  ubhzEFnTIZd+50xx+7LSYK05qAvqFyFWhfFQDlnrzuBZ6brJFe+GnY+EgPbk6ZGQ  3BebYhtF8GaV0nxvwuo77x/Py9auJ/GpsMiu/X1+mvoiBOv/2X/qkSsisRcOj/KK  NFtY2PwByVS5uCbMiogziUwthDyC3+6WVwW6LLv3xLfHTjuCvjHIInNzktHCgKQ5  ORAzI4JMPJ+GslWYHb4phowim57iaztXOoJwTdwJx4nLCgdNbOhdjsnvzqvHu7Ur  TkXWStAmzOVyyghqpZXjFaH3pO3JLF+l+/+sKAIuvtd7u+Nxe5AW0wdeRlN8NwdC  jNPElpzVmbUq4JUagEiuTDkHzsxHpFKVK7q4+63SM1N95R1NbdWhscdCb+ZAJzVc  oyi3B43njTOQ5yOf+1CceWxG1bQVs5ZufpsMljq4Ui0/1lvh+wjChP4kqKOJ2qxq  4RgqsahDYVvTH9w7jXbyLeiNdd8XM2w9U/t7y0Ff/9yi0GE44Za4rF2LN9d11TPA  mRGunUHBcnWEvgJBQl9nJEiU0Zsnvgc/ubhPgXRR4Xq37Z0j4r7g1SgEEzwxA57d  emyPxgcYxn/eR44/KJ4EBs+lVDR3veyJm+kXQ99b21/+jh5Xos1AnX5iItreGCc=  -----END CERTIFICATE-----  )EOF";  void reconnect() {  while (!client.connected()) {  Serial.print("Attempting MQTT connection… ");  String clientId = "ESP32Client";  if (client.connect(clientId.c\_str(), mqtt\_username, mqtt\_password)) {  Serial.println("connected!");  } else {  Serial.print("failed, rc = ");  Serial.print(client.state());  Serial.println(" try again in 5 seconds");  delay(5000);  }  }  }  void setup\_wifi() {  delay(10);  Serial.println();  Serial.print("Connecting to ");  Serial.println(ssid);  WiFi.mode(WIFI\_STA);  WiFi.begin(ssid, password);  while (WiFi.status() != WL\_CONNECTED) {  delay(500);  Serial.print(".");  }  Serial.println("");  Serial.println("WiFi connected");  Serial.println("IP address: ");  Serial.println(WiFi.localIP());  }  void setup() {  delay(500);  Serial.begin(9600);  delay(500);  Esp32GpsSerial.begin(9600);  delay(500);  setup\_wifi();  espClient.setCACert(root\_ca);  client.setServer(mqtt\_server, mqtt\_port);  client.setCallback(callback);  if (!client.connected()) {  reconnect();  }    } |
| \_05imi\_mqttcloud\_loop  Arduino Cloud雲端平台 loop區設定 | 'client.loop();' |
| \_06imi\_mqttcloud\_submqtt  setup()  CAR\_GPS  [訂閱]MQTT主題 | //訂閱  client.subscribe(TOPIC\_CAR\_GPS);  Serial.println(String(TOPIC\_CAR\_GPS)+"已訂閱成功！請開啟https://console.hivemq.cloud/，並用Publish Message來測試！"); |
| \_07imi\_mqttcloud\_mqttcallback\_func  MQTT回覆函式  中區：statements\_msg | //MQTT訂閱的主題回覆  void callback(char\* topic, byte\* payload, unsigned int length) {  //傳送至ESP32 GPS板(格式:xy)  msgXY[0] = (char)payload[0];  msgXY[1] = (char)payload[1];  msgXY[2] = \'\\0\';  Serial.println(msgXY);    Statements區塊  } |
| \_08imi\_mqttcloud\_sendarduino\_gps  [傳送]訊息至ESP32 GPS板:地圖xy座標 | 'Esp32GpsSerial.write(msgXY);' |
|  |  |
|  |  |
|  |  |