#include <WiFi.h>

#include <WiFiClient.h>

#include <PubSubClient.h>

#include <ArduinoJson.h>

#include<TinyGPS++.h>

#define RXD2 16

#define TXD2 17

HardwareSerial neogps(1);

TinyGPSPlus gps;

Char arr[100];

Const char\* ssid = “Redmi”;

Const char\* password = “krish@08”;

#define ID “17cmwk”

#define DEVICE\_TYPE “Tracker”

#define DEVICE\_ID “gps1”

#define TOKEN “childtracker1”

Char server[] = ID “.messaging.internetofthings.ibmcloud.com”;

Char publish\_Topic1[] = “iot-2/evt/Data1/fmt/json”;

Char publish\_Topic2[] = “iot-2/evt/Data2/fmt/json”;

Char authMethod[] = “use-token-auth”;

Char token[] = TOKEN;

Char clientId[] = “d:” ID “:” DEVICE\_TYPE “:” DEVICE\_ID;

WiFiClient wifiClient;

PubSubClient client(server, 1883, NULL, wifiClient);

Void setup() {

Serial.begin(115200);

Serial.println();

Wifi\_init();

}

Long previous\_message = 0;

Void loop() {

Client.loop();

String payload = getLocationPayload();

If(payload==”{}”){

Return;

}

Serial.print(“Sending payload: “);

Serial.println(payload);

If (client.publish(publish\_Topic1, arr)) {

Serial.println(“Published successfully”);

} else {

Serial.println(“Failed”);

}

Delay(2000);

}

Void wifi\_init(){

WiFi.begin(ssid, password);

Neogps.begin(9600,SERIAL\_8N1,RXD2,TXD2);

While (WiFi.status() != WL\_CONNECTED) {

Delay(500);

Serial.print(“.”);

}

Serial.println(“”);

Serial.println(WiFi.localIP());

If (!client.connected()) {

Serial.print(“Reconnecting client to “);

Serial.println(server);

While (!client.connect(clientId, authMethod, token)) {

Serial.print(“.”);

Delay(500);

}

Serial.println(“Connected TO IBM IoT cloud!”);

}

}

String getLocationPayload(){

Boolean newData = false;

For(unsigned long start = millis();millis()-start<1000;){

While(neogps.available()){

If(gps.encode(neogps.read())){

newData = true;

}

}

}

String payload;

If(newData == true){

newData = false;

payload = locationPayloadGenerator();

}

Else{

Serial.println(“No data”);

Payload =”{}”;

}

Return payload;

}

String locationPayloadGenerator(){

String payload = “{}”;

If(gps.location.isValid()){

Float lat = gps.location.lat();

Float lon = gps.location.lng();

Payload = “{\”latitude\” : “+String(lat)+”,\”longitude\” : “+String(lon)+”}”;

Create\_json(lat,lon);

}

Return payload;

}

Void create\_json(float lat,float lon){

StaticJsonDocument<100> doc;

JsonObject root = doc.to<JsonObject>();

Root[“name”]=”Child”;

Root[“latitude”] = lat;

Root[“longitude”] = lon;

serializeJsonPretty(doc,arr);

}