# Sprint-4

| | |
|---|---|
| Date | 16 November 2022 |
| Team ID | PNT2022TMID21568 |
| Project Name | Personal Assistance for Seniors Who AreSelf Reliant |

**Code :**

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include <LiquidCrystal_I2C.h>
#include <DHT.h>// Library for dht11
#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT11   // define type of sensor DHT 11
#define LED 2
DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
```

```
//-------credentials of IBM Accounts------

#define ORG "7kzrri"//IBM ORGANITION ID
#define DEVICE_TYPE "nodemcu"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "12345"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678"     //Token
String data3="";
int buzz= 13;

//-------- Customise the above values --------
char server[] = ORG".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/type/nodemcu/id/12345/evt/status/fmt/json ";// topic name and type of event perform and format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
LiquidCrystal_I2C lcd(0x27,32,2);
```

```
//---------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing parameter like server id,portand wificredential

void setup()// configureing the ESP32
{
  Serial.begin(115200);
  dht.begin();
  pinMode(buzz, OUTPUT);
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
}
void loop()// Recursive Function
{
  if (!client.loop()) {
    mqttconnect();
  }
}
```

```
/.....retrieving to Cloud...../
void mqttconnect() {
 if (!client.connected()) {
  Serial.print("Reconnecting client to ");
  Serial.println(server);
  while (!!!client.connect(clientId, authMethod, token)) {
   Serial.print(".");
   delay(500);
  }

   initManagedDevice();
   Serial.println();
 }
}
void wificonnect() //function defination for wificonnect
{
 Serial.println();
 Serial.print("Connecting to ");
 WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
 while (WiFi.status() != WL_CONNECTED) {
  delay(500);
```

```
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
void initManagedDevice() {
  if (client.subscribe(subscribetopic)) {
    Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
}
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);
  for (int i = 13; i < payloadLength-2; i++) {
    //Serial.print((char)payload[i]);
```

```
   data3 += (char)payload[i];
  }
  Serial.println("Medicine Name: "+ data3);
  if(data3 != "")
  {
   lcd.init();
   lcd.print(medicine);
   digitalWrite(LED,HIGH);
   tone(buzz, 100, 1000);
   delay(2000);
   digitalWrite(LED,LOW);
  noTone(buzz);
  delay(1000);
  }
  else
  {
digitalWrite(LED,LOW);
  }
data3 ="";
}
```

**Output:**

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include <LiquidCrystal_I2C.h>
#include "DHT.h"// Library for dht11
#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT11   // define type of sensor DHT 11
#define LED 2
DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dh
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);


//-------credentials of IBM Accounts------

#define ORG "64yf7x"//IBM ORGANITION ID
#define DEVICE_TYPE "b11m3edevicetype"//Device type mentioned in ibm watson IOT
#define DEVICE_ID "b11m3edeviceid"//Device ID mentioned in ibm watson IOT Platfo
#define TOKEN "-&EMtr7l-v-G2zG))e"      //Token
String data3="";
int buzz= 13;

//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
LiquidCrystal_I2C lcd(0x27,16,2);

//-----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefine

void setup()// configureing the ESP32
{
```