THE HONG KONG POLYTECHNIC UNIVERSITY
香港理工大學

COLLEGE OF PROFESSIONAL AND CONTINUING EDUCATION
專業及持續教育學院

School of Professional Education and Executive Development
專業進修學院
SPEED

Seminar on Overview of the Financial and Business Markets in Hong Kong and the Applications of Artificial Intelligence Technology

Dr James CHAU

19 January 2026

Topic

1. Wireless IoT Technologies and Applications
2. Analysis based on the Financial Market in Hong Kong
3. Business Model and Innovation Case analysis
4. Applications and Practical Implementation of Artificial Intelligence
5. Group Presentation and Closing Ceremony

Lecturer information

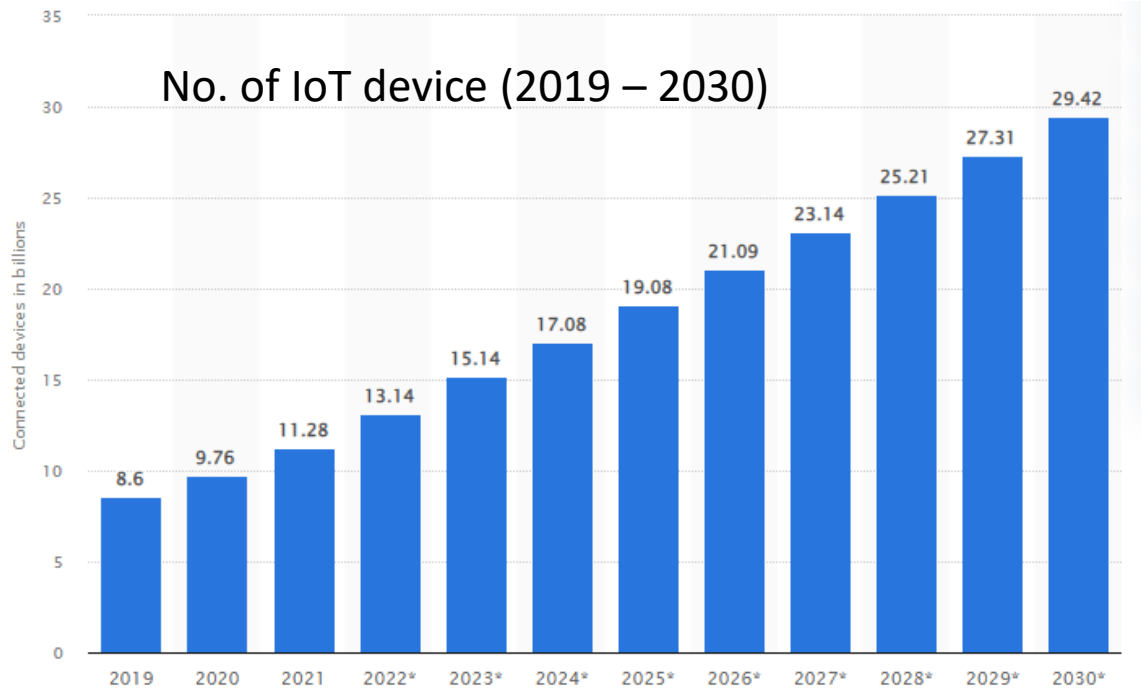
- Dr. Chau Chun Pong, James
  - PhD in Electronic and Information Engineering, PolyU
  - Principal Lecturer of CPCE, PolyU
  - Associate Division Head of SEHS, CPCE, PolyU
- Research Interests:
  - Image processing, video processing, pattern recognition, artificial intelligence, Integrated circuits, indoor positioning, robotics.
- Tel: 3746 0237
- Email: ccjames@cpce-polyu.edu.hk

Day 1: Wireless IoT Technologies and Applications

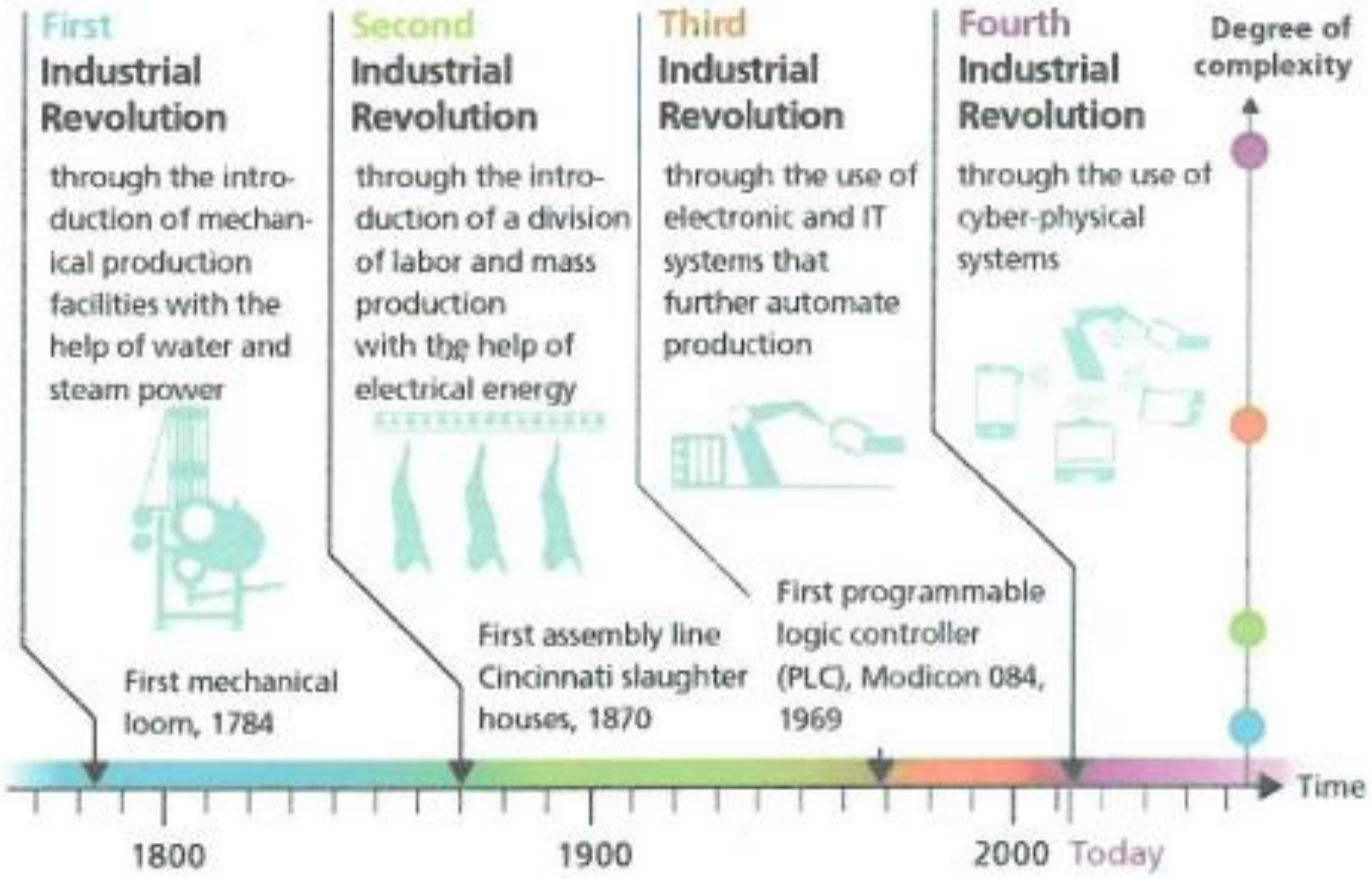
- Introduction to IoT
- Wireless IoT communication technologies
- IoT applications in various industries
- IoT device practice
- Conclusion and Q&A

What is Internet of Things (IOT)?

- It means things can connect to Internet.
  - How? Through the control of computer systems
- Many companies starts developing their products in IOT
  - Intel, IBM, Microsoft, ARM, CISCO…
  - In 2030, over 29 billion devices are connected to Internet.
- Major technologies:
  - Embedded systems
  - Big data
  - RFID
  - Sensors
  - Actuators



Industrial revolution



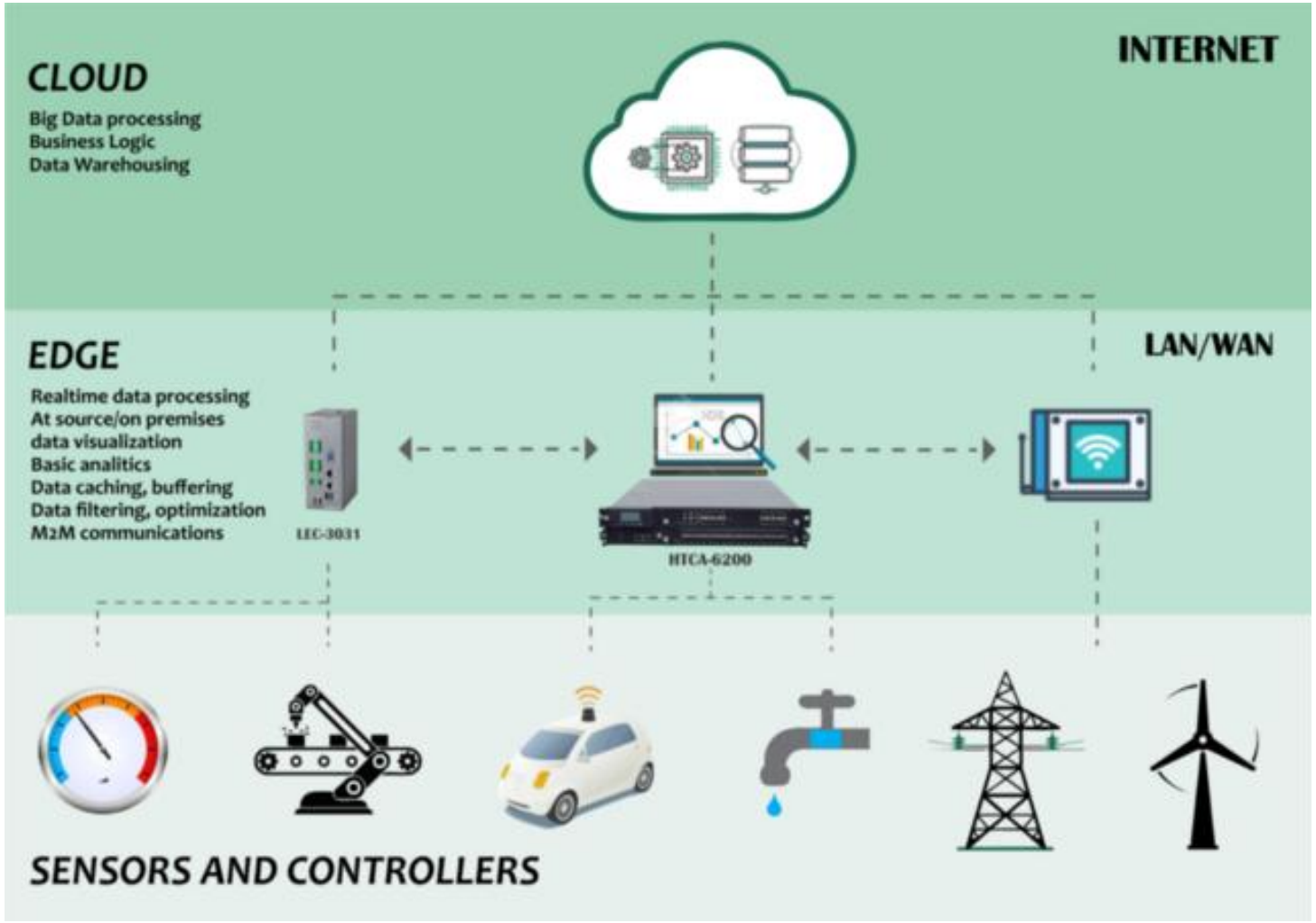
Smart City Development in Hong Kong

- Blueprint: https://www.smartcity.gov.hk/
- Objectives:
  - make use of innovation and technology to address urban challenges, enhance the effectiveness of city management and improve people's quality of living as well as Hong Kong's sustainability, efficiency and safety;
  - Enhance Hong Kong's attractiveness to global businesses and talents; and
  - Inspire continuous city innovation and sustainable economic development
- Areas:
  - Smart Mobility, Smart Living, Smart Environment, Smart People, Smart Government and Smart Economy

Example

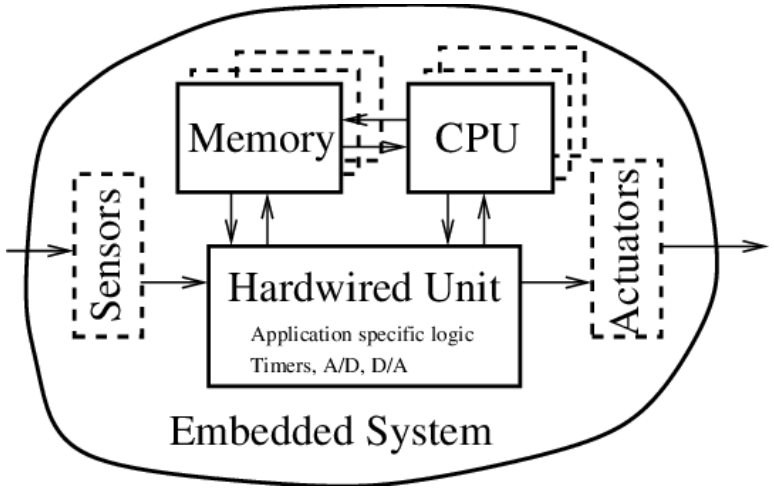
https://youtu.be/LlhmzVL5bm8

IOT model


CLOUD
Big Data processing
Business Logic
Data Warehousing
INTERNET
EDGE
Realtime data processing
At source/on premises
data visualization
Basic analitics
Data caching, buffering
Data filtering, optimization
M2M communications
LEC-3031
HTCA-6200
LAN/WAN
SENSORS AND CONTROLLERS

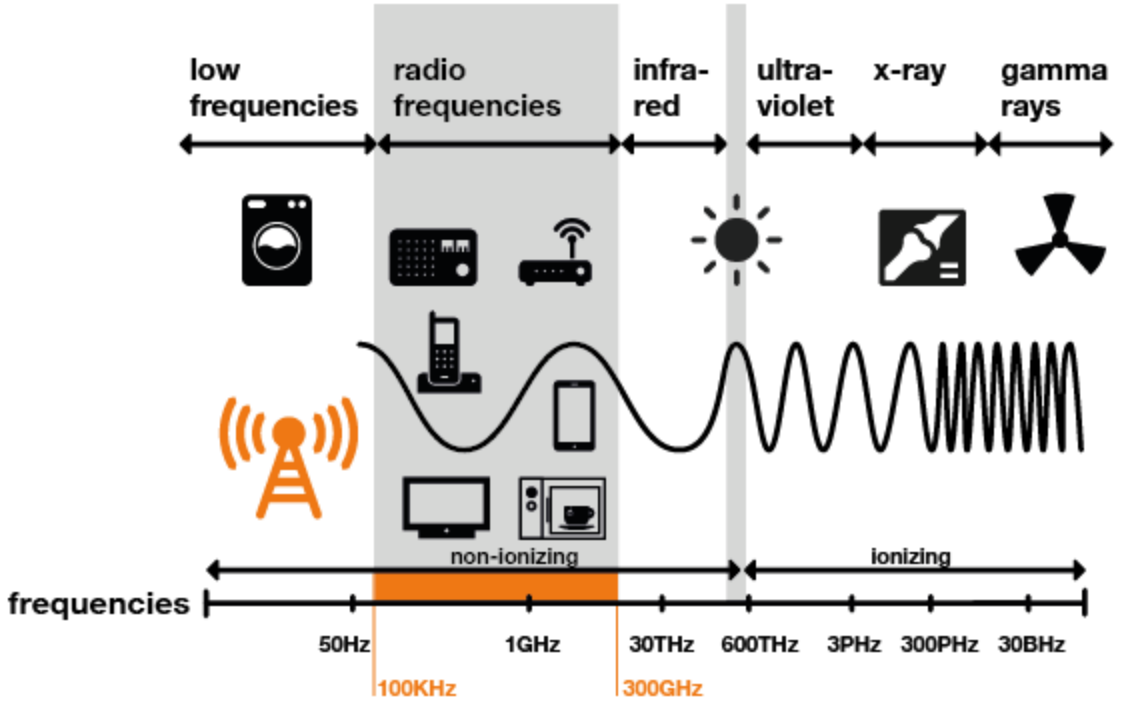
Embedded system

- An embedded system is a combination of computer hardware and software, either fixed in capability or programmable, designed for a specific function or functions within a larger system.
- Nowadays, they are in hand-held size, light-weighted and cheap.
- Examples:
  - Nvidia Jetson
  - Raspberry Pi
  - Micro:bit
  - Arduino
  - ESP8266



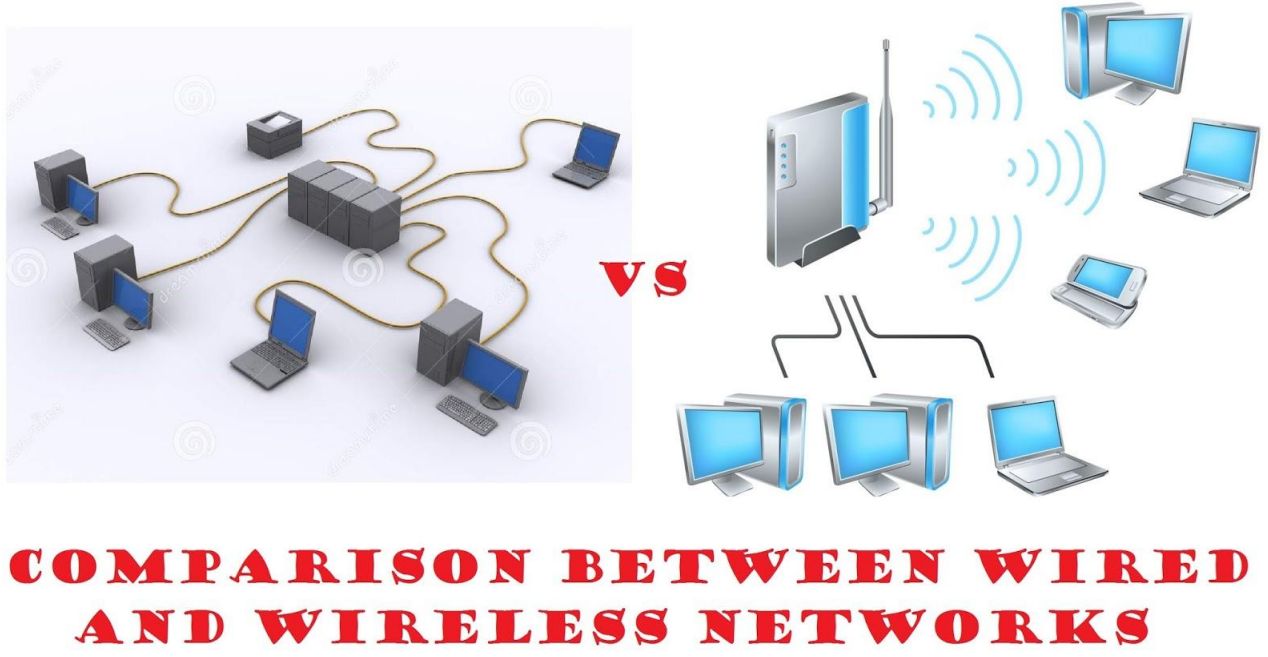
Electronic communication

- Electronic communication means the transmission of data/messages between two or more locations
- Nowadays, electromagnetic radiations are employed in the transmission
- The messages and data are converted into their digital form for processing and transmission



Wire and wireless communication

- Wire communication
  - Physic connection between the transmitter and receiver is existed in terms of cable
  - E.g. Optical fibres
  - More reliable
- Wireless communication
  - The transmission medium is air
  - E.g. mobile communication network
  - More convenience



Wireless IoT communication technologies

- Short-range communication technologies
  - Range are determined by transmission power, short-range transmission devices often transmits signals using low power
  - Transmission distance is around 1 m to 10 m
  - Examples
    - Bluetooth and Bluetooth Low Energy (BLE)
    - Wi-Fi (IEEE 802.11)
    - Zigbee (IEEE 802.15.4)
    - Z-Wave
    - NFC (Near Field Communication)

Bluetooth and Bluetooth Low Energy (BLE)

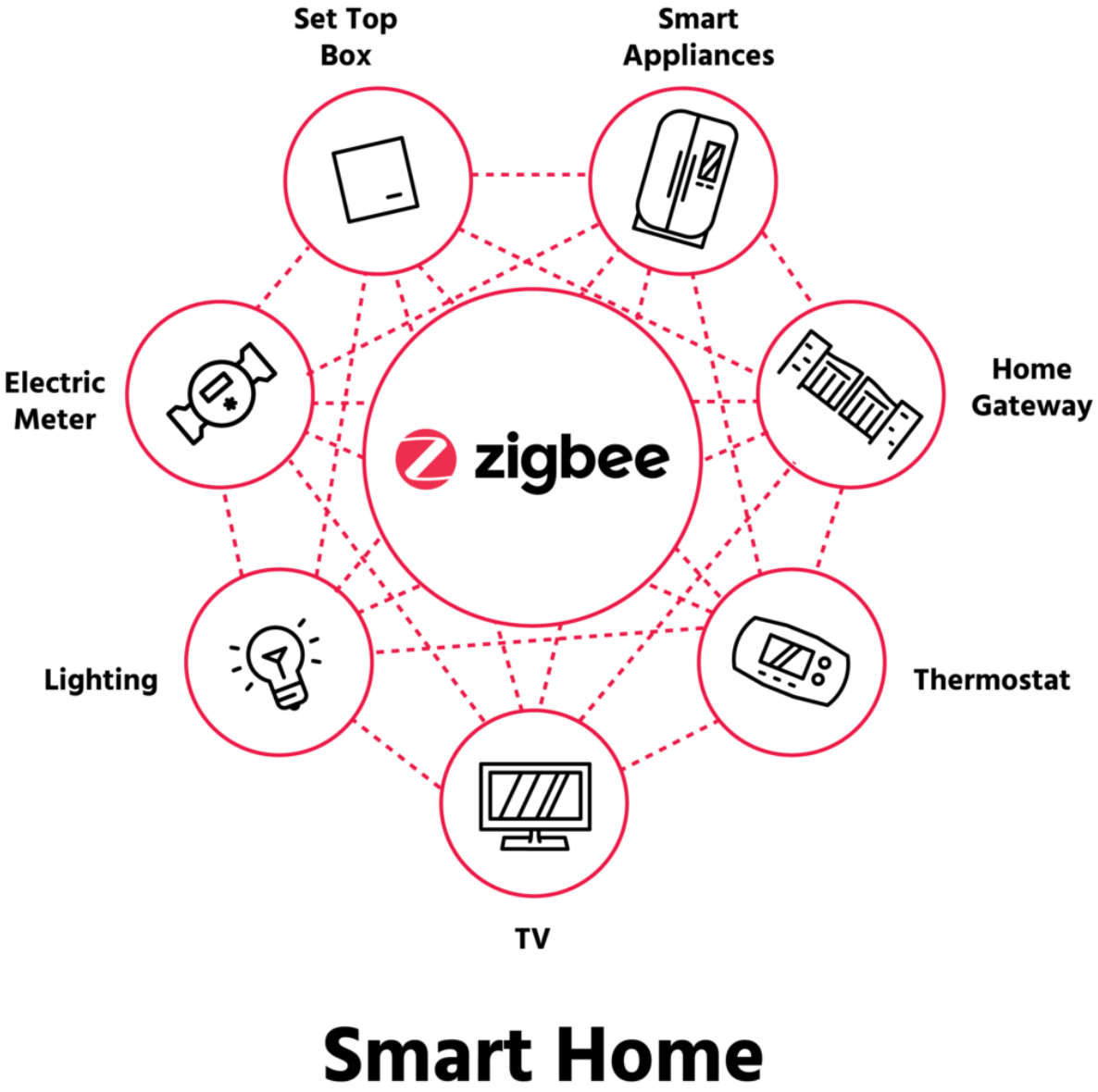
- Bluetooth is now a wireless standard transmission technology
- It enables communication between devices
- BLE also known as Bluetooth Smart
  - It is a power-efficient version of Bluetooth technology
- Current version is 6.x, transmission range can be up to 240 m and the maximum transmission rate is 100 Mbps, its transmission frequency is 2.4 GHz.

Wi-Fi

- Wi-Fi = Wireless Fidelity
- It is a wireless networking technology which allows the devices connect to the Internet
- It uses radio waves with frequency of 2.4 GHz and 5 GHz for transmission
- Current version = version 6 (802.11 ax), its maximum transmission rate is 9.6 Gbps

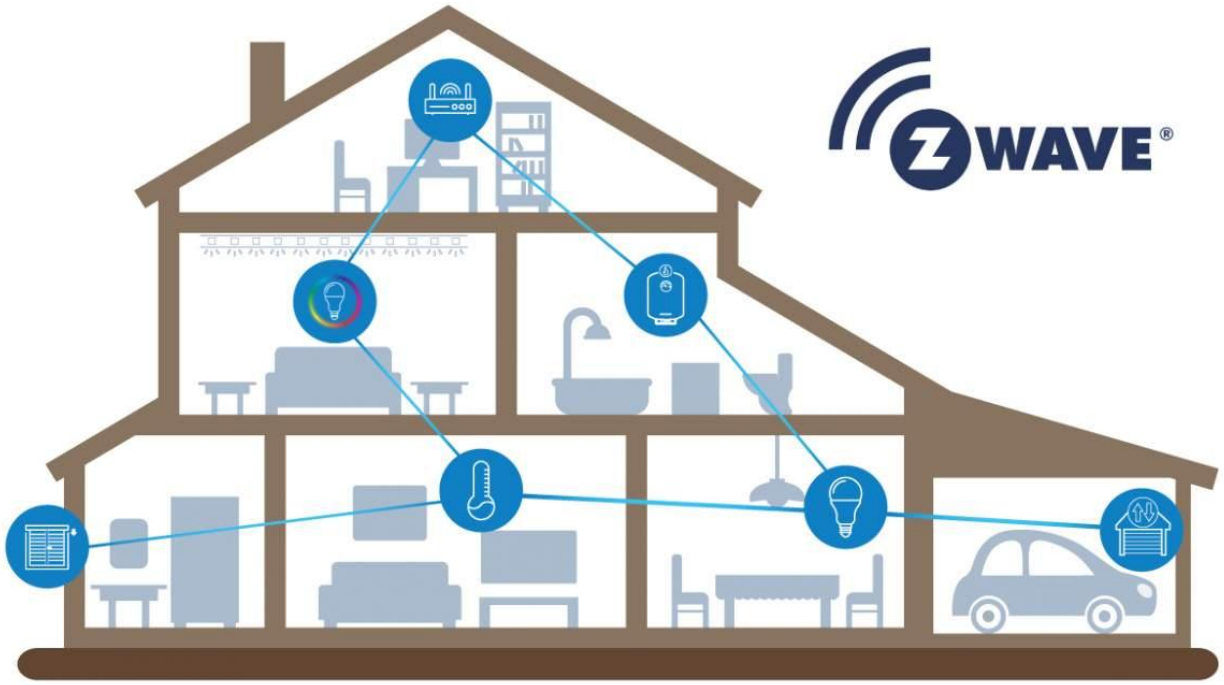
Zigbee

- Zigbee is a wireless communication protocol specifically designed for low-power, low-data-rate wireless applications
- It operates again in 2.4 GHz
- Applications: home automation, smart lighting, industrial control systems and building automation
- It supports mesh networking and can easily extend its range



Z-wave

- Z-wave is a wireless communication protocol for how automation and smart home devices
- It aims at providing reliable and secure communication for the devices
- It operates in 868.42 MHz or 908.42 MHz and has better signa penetration and it supports also mesh network



Near Field Communication (NFC)

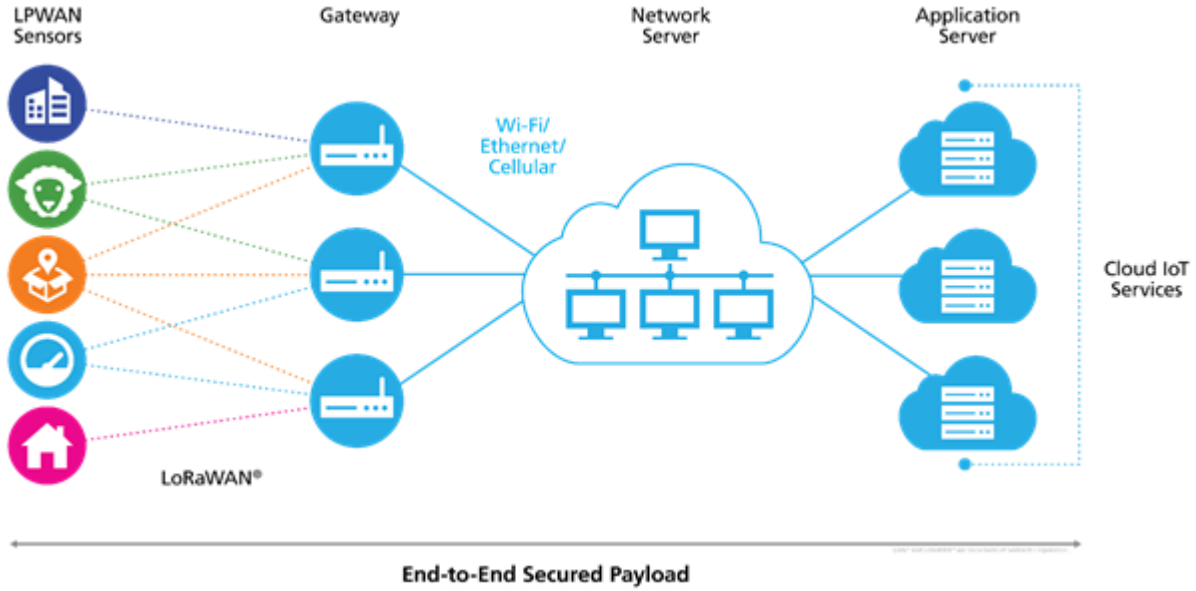
- NFC is a short-range wireless communication technology which transmit data only when the devices are brought in close proximity, usually a few centimetres
- It operates in 13.56 MHz and uses Radio Frequency Identification (RFID) technology
- It has already used in some contactless payment systems or device pairing up

Long-range communication technologies

- Compared to Short-range communication technologies, Long-range communication technologies can transmit the messages between longer distances with a higher power in transmission
- Examples:
  - Long Range Wide Area Network (LoRa WAN)
  - Sigfox
  - NB-IoT (Narrowband IoT)
  - LTE-M (Long-Term Evolution for machines)

Long Range Wide Area Network

- LoRaWAN is a low power, long range wireless communication protocol and it is widely applied in IoT applications
- It employs 900 MHz for data transmission with good penetration ability (up to 10 km distance) but the data rates is only 50 kbps

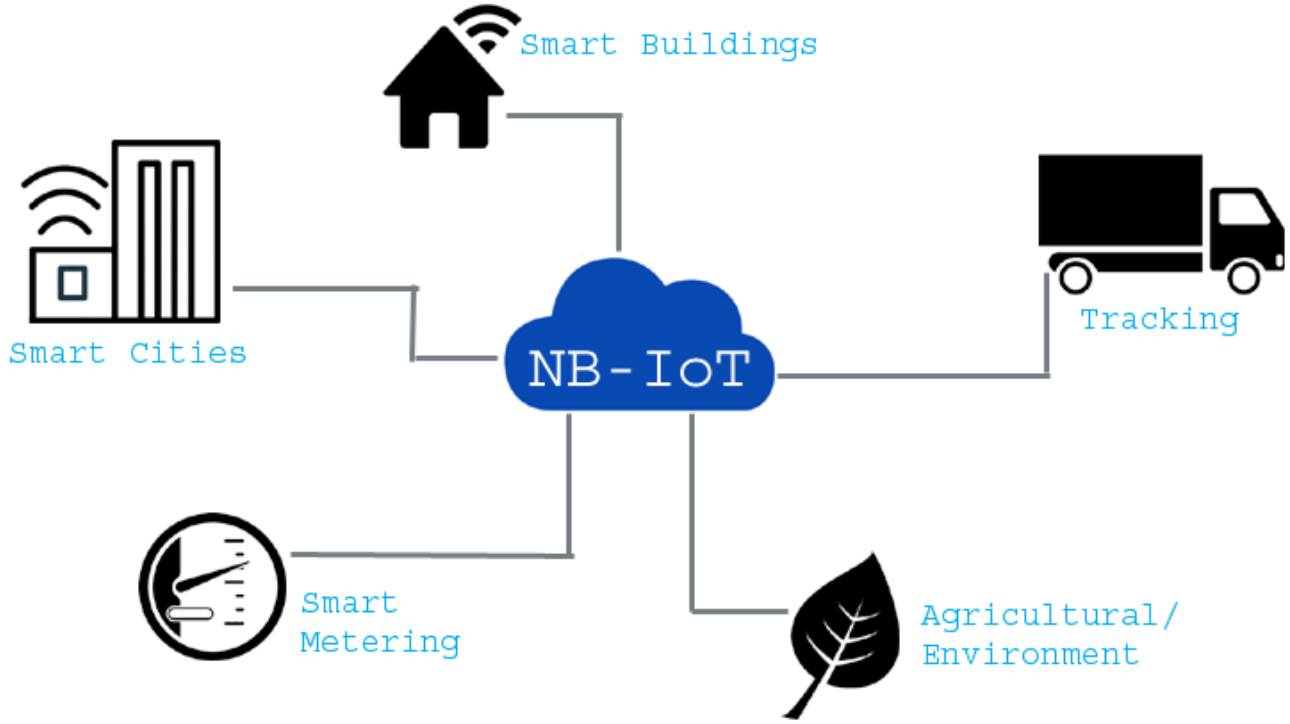


Sigfox

- Sigfox is another lower power WAN technologies and it is suitably applied to IoT applications
- It employs 868 MHz or 902 MHz for data transmission, but the data rate is only 100 bps to 1 kbps only
- It can transmit data with a range up to 40 km and can reach the underground devices

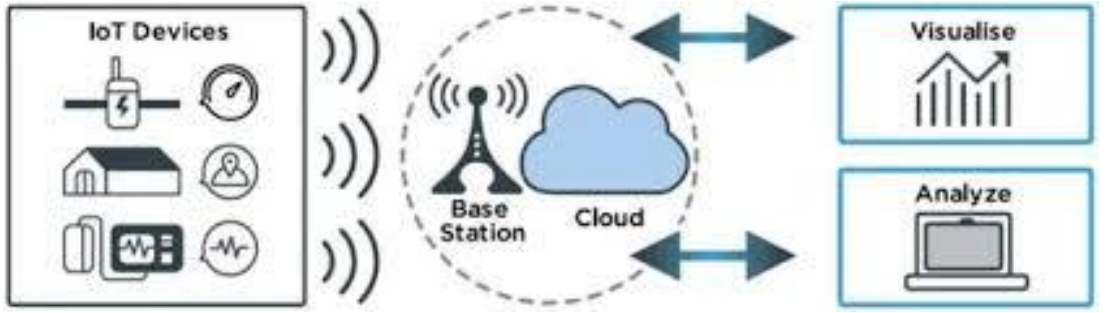
Narrowband-IoT

- NB-IoT is a cellular network technology designed for IoT applications
- It uses the LTE frequency bands with a narrower bandwidth (200 kHz) for data transmission
- It is targeted the indoor IoT devices and its transmission range can be up to 10 km



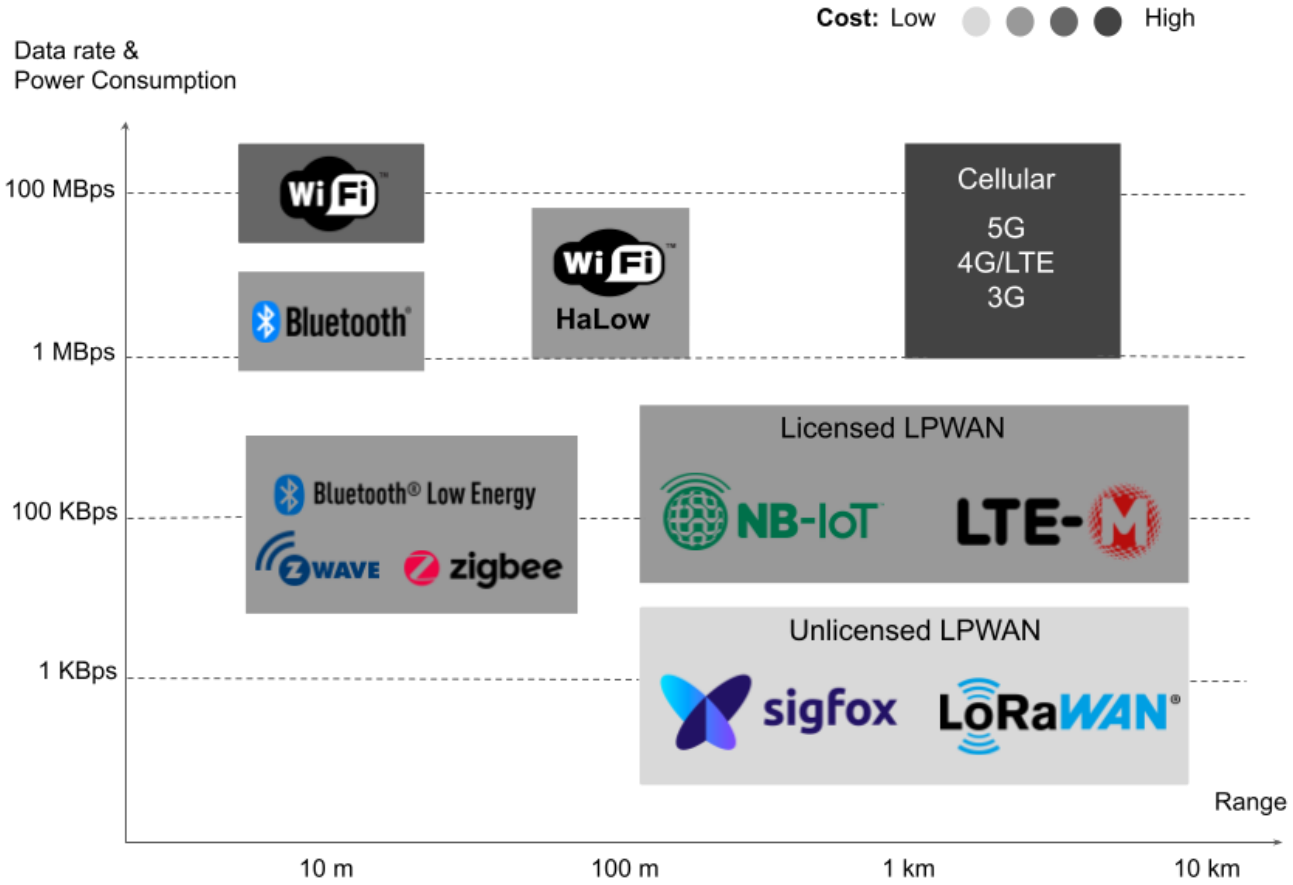
Long-Term Evolution for Machines

- LTE-M is a cellular wireless standard beyond 3G. When it is applied to machines, it can provide a high data rate (up to 100 Mbps)
- It can use any transmission frequency available in LTE ranged from 600 MHz to 5.2 GHz
- As it can transmit in a higher data rate, it makes the power consumption of the devices become higher



Wireless communication technology selection criteria

- Range
- Power consumption
- Data rates
- Network scalability



IoT applications (Smart home)

- Home automation
- Energy management
- Security and surveillance

IoT applications (Industrial IoT)

- Predictive maintenance
- Asset tracking
- Process automation

IoT applications (Smart Cities)

- Traffic management
- Waste management
- Public safety

IoT applications (Healthcare)

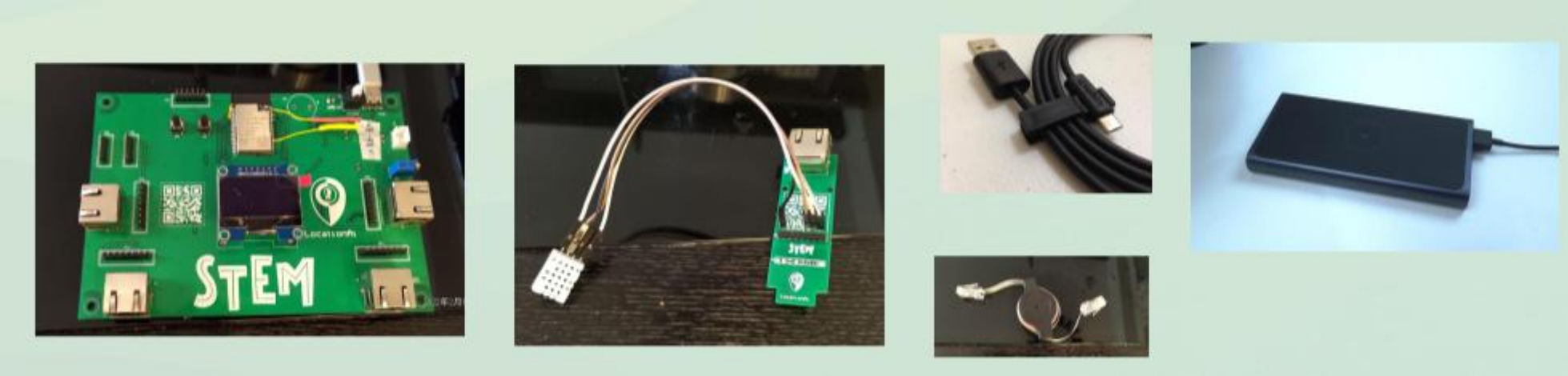
- Remote patient monitoring
- Telemedicine
- Wearable devices

IoT applications (Agriculture)

- Precision farming
- Livestock monitoring
- Irrigation management

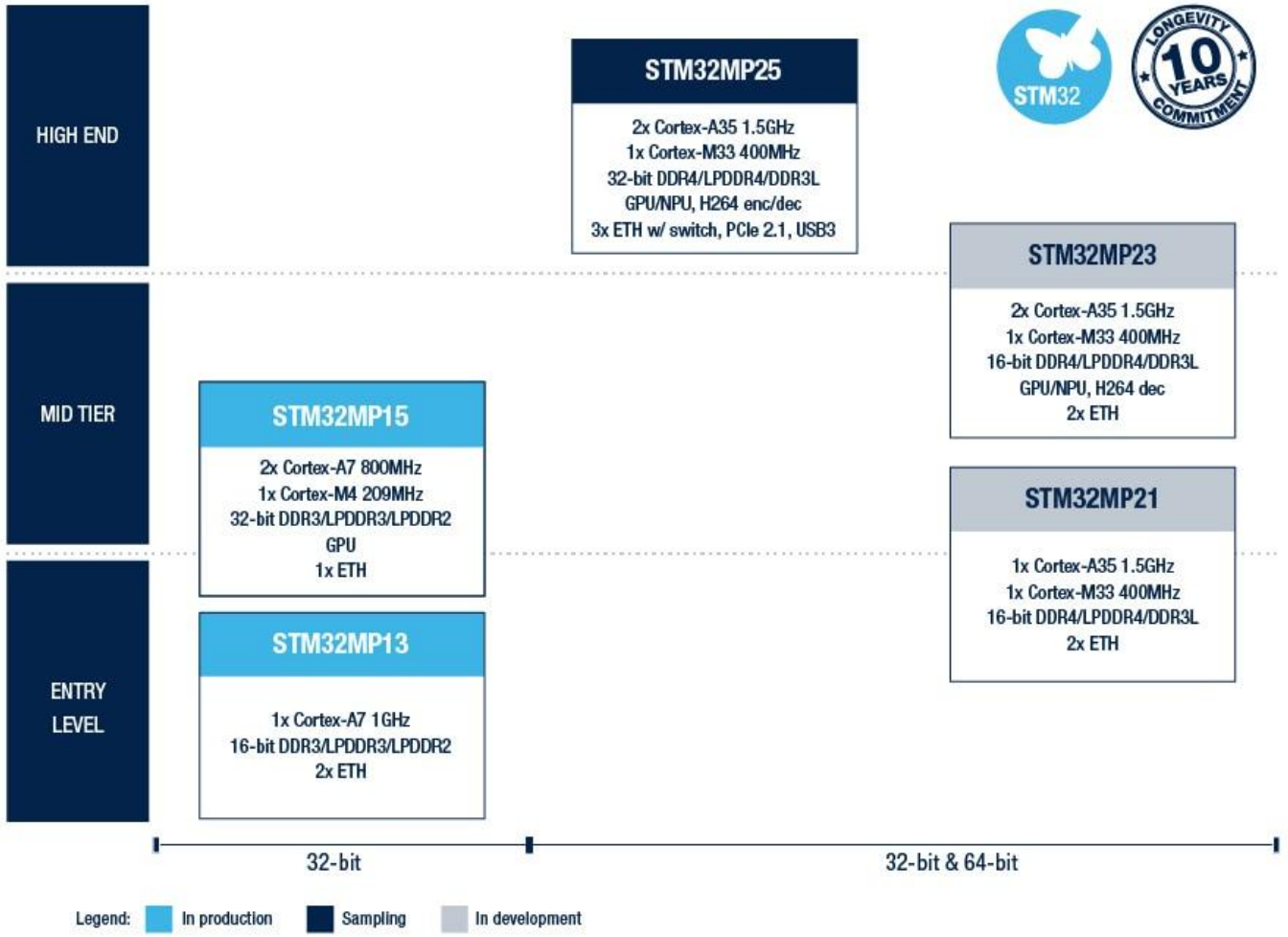
IoT practice – Smart Location IoT kit

- MCU produced by STMicroelectronics
- Modules that integrated with the other components and sensors
- Objective: A low-cost WiFI microchip with built-in networking software and microcontroller capability

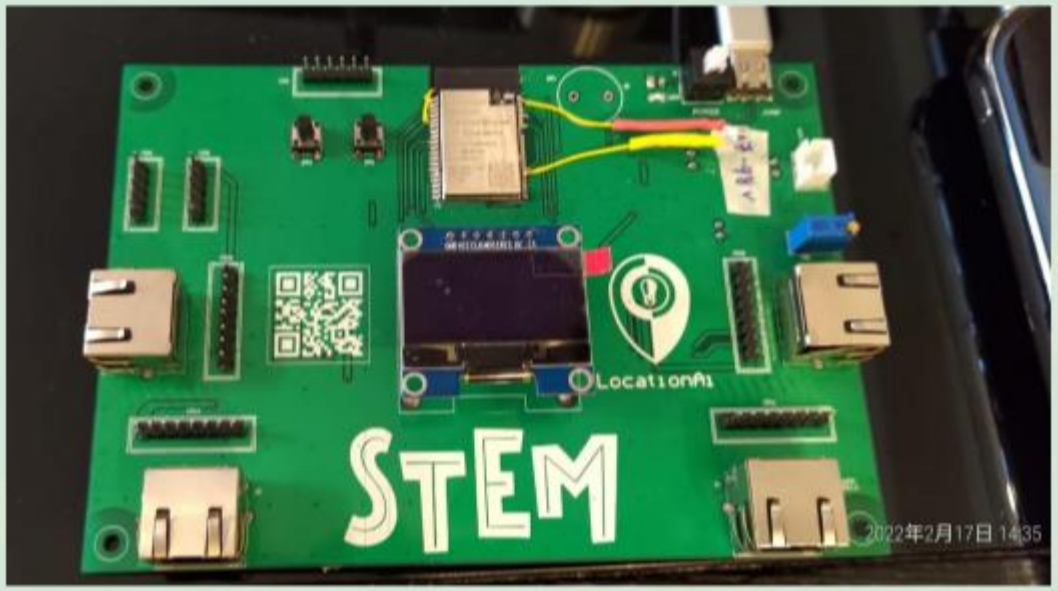


STMicroelectronics processors

- Arm Cortex MPUSs



Hardware package



- A standard PCB is a plastic card covered with fiberglass.
  - Various important components are mounted on this non-conductive plate
  - connected through small paths called traces.
  - These traces allow electrical components on the circuit board to function by conveying electrical information.
  - The PCB also has small holes drilled into it to connect each component.

- In the center of the PCB, there is an LED display that displays information and results from any connected sensors. The information displayed is programmable.

Hardware package

2. Socket Wire - Socket cable is the transmission medium between the PCB circuit board and the IoT sensor

3. Temperature and humidity sensor

4. CO2 + TVOC (Total volatile organic compounds) sensor

5. USB Type C wire - Powers the PCB

How to assemble?

1. Connect sensors with circuit wires and plug them to PCB
2. Plug the USB cable into the circuit board and power bank (or computer USB plug)
3. Wait 20 seconds, you will see Wi-Fi connected on the LED display
4. Check the LED display
5. Humidity and temperature + CO_2 value will be shown

Data Analysis

How the hardware suite operates, collects and uploads data

- The IoT sensor will sense the ambient temperature, humidity and CO_2, then transmit the data through PCB.
- The PCB board contains a Wi-Fi communication module. When the IoT sensor is connected to the PCB board, the data will be output to the Internet through the Wi-Fi module.
- The cloud server will receive the ambient data in real time and then store it in the database.
- Data will be collected and visualized in management platform.

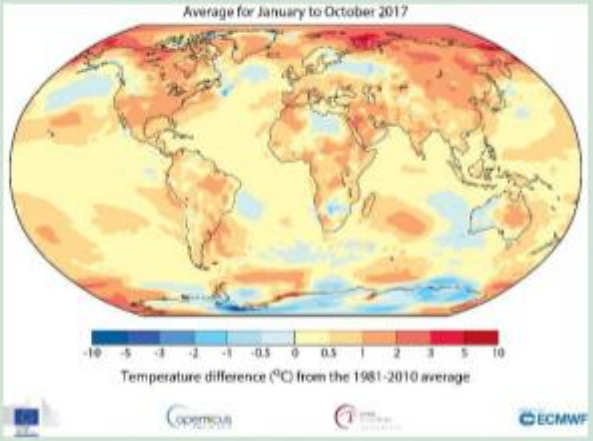
Data analysis application

- IAQ – Indoor air quality
- The impact of indoor CO_2 concentration to human

CO_2 [ppm]	Air Quality
2100	BAD Heavily contaminated indoor air Ventilation required
2000	
1900	
1800	
1700	
1600	
1500	MEDIOCRE Contaminated indoor air Ventilation recommended
1400	
1300	
1200	
1100	
1000	FAIR
900	
800	GOOD
700	
600	EXCELLENT
500	
400	

Temperature Guide

°C	°F	Human
-20	-4	Extreme Cold
-10	14	Very Cold
0	32	Cold
10	50	Cool
20	68	Brisk
30	86	Warm
35	95	Hot
40	104	Very Hot
50	122	Exteme Heat



Light Sensor

Temperature Sensor

Gas Sensor

Dust Sensor

Grove environmental sensors

3-Axis Digital Accelerometer (±16g)

3-Axis Digital Compass

3-Axis Analog Accelerometer

3-Axis Digital Accelerometer (±1.5g)

Grove motion sensors

433MHz Simple RF Link Kit

Serial Bluetooth

Infrared Receiver

GPS

Grove wireless modules

Thumb Joystick

Solid State Relay

OLED Display 96*96

I2C Motor Driver

Grove user interface modules

Ear-clip Heart Rate Sensor

PIR Motion Sensor

Alcohol Sensor

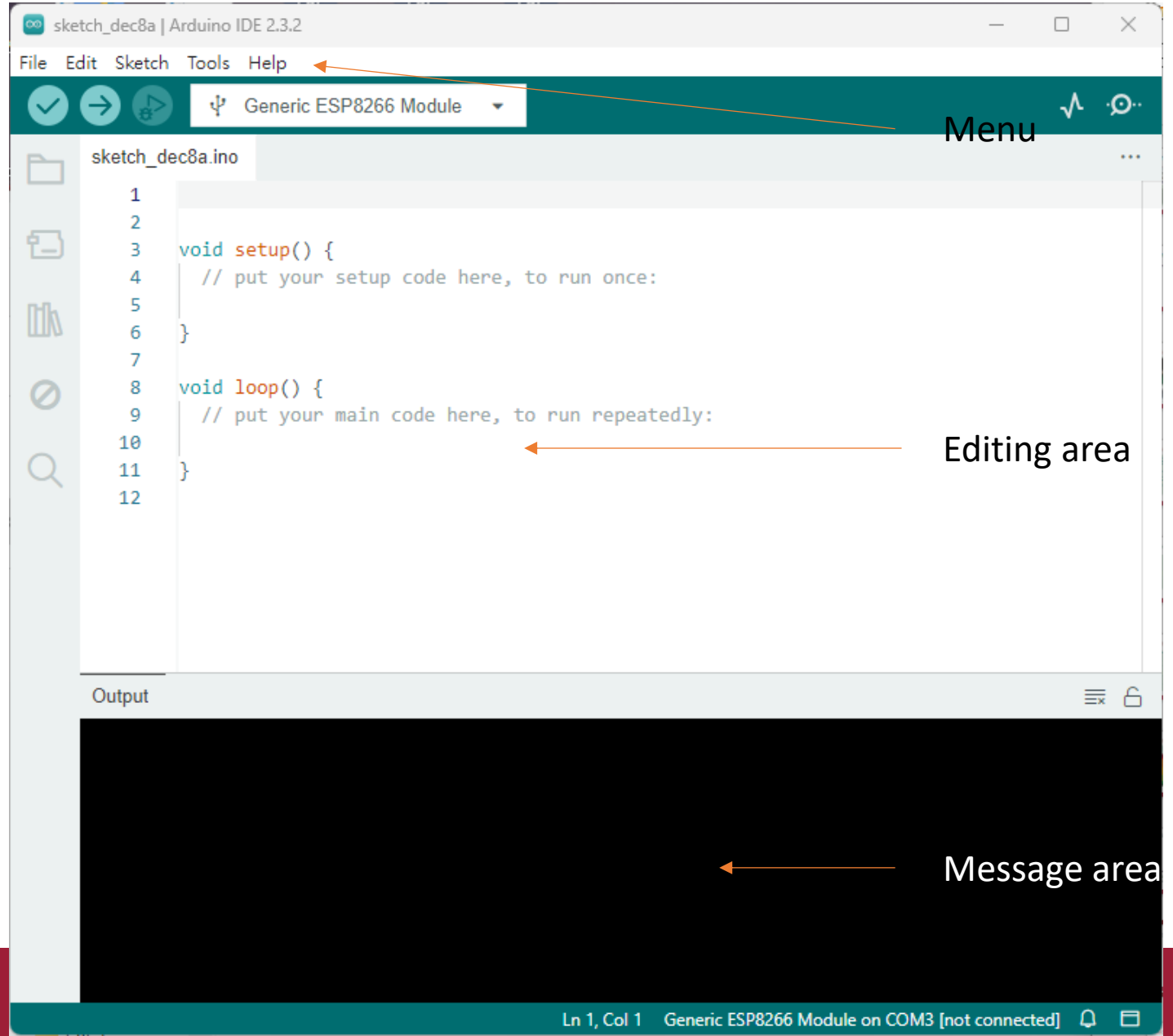
I2C Color Sensor

Grove physical sensors

Arduino Integrated Development Environment

- Download latest IDE software at Arduino.cc: https://arduino.cc/
- Unzip the file to an appropriate location
- Run Arduino.exe

Arduino IDE

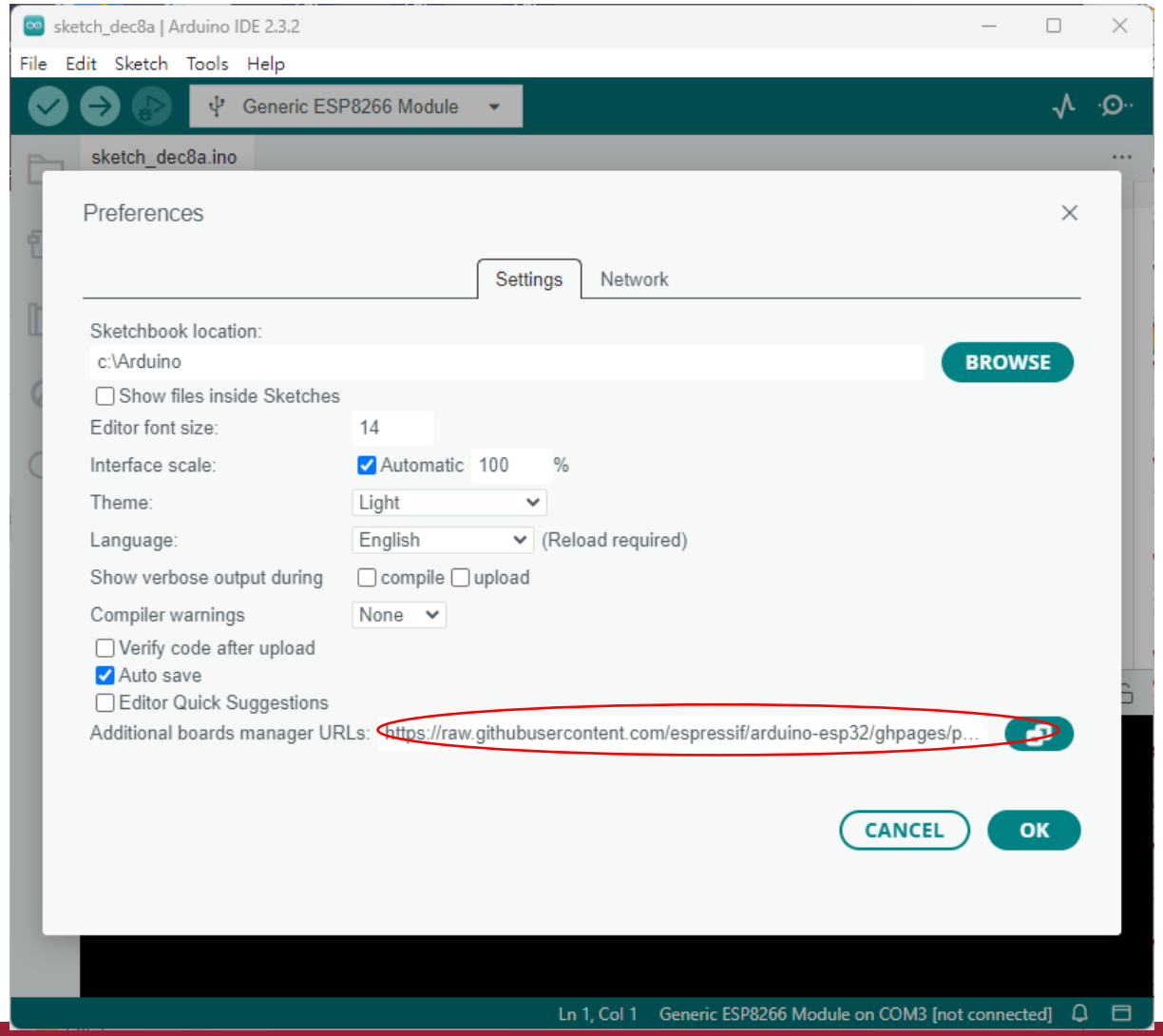

sketch_dec8a | Arduino IDE 2.3.2
File Edit Sketch Tools Help
Generic ESP8266 Module
Menu
sketch_dec8a.ino
1 2 3 4 5 6 7 8 9 10 11 12
void setup() {
// put your setup code here, to run once:
}
void loop() {
// put your main code here, to run repeatedly:
}
Editing area
Output
Message area
Ln 1, Col 1 Generic ESP8266 Module on COM3 [not connected]

Arduino program structure

- An Arduino program is known as sketch.
- It consists of two parts
  - **Setup function**: it contains a set of instructions for the Arduino to execute once only, each time it is reset or turned on.
  - **Loop function**: it tell the Arduino to execute an instruction over and over until the power is shut off or the reset button is pressed.

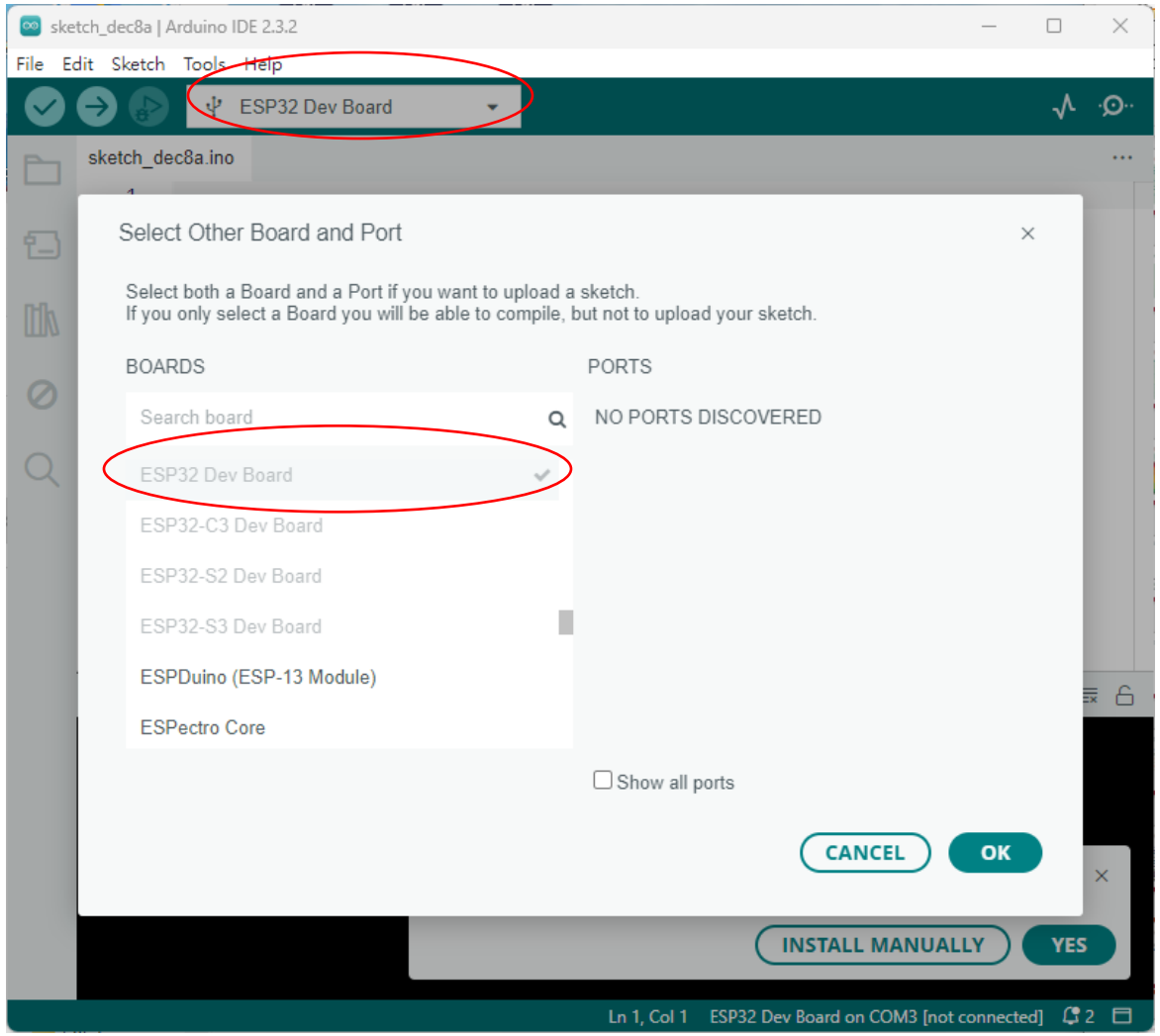
Configuration for Arduino ide

- Steps:
  1. Start the **Arduino IDE** and open the **Preferences** window under the File menu.
  2. Enter the following URL into the **Additional Board Manager** URLs field: https://raw.githubusercontent.com/espressif/arduino-esp32/ghpages/package_esp32_index.json

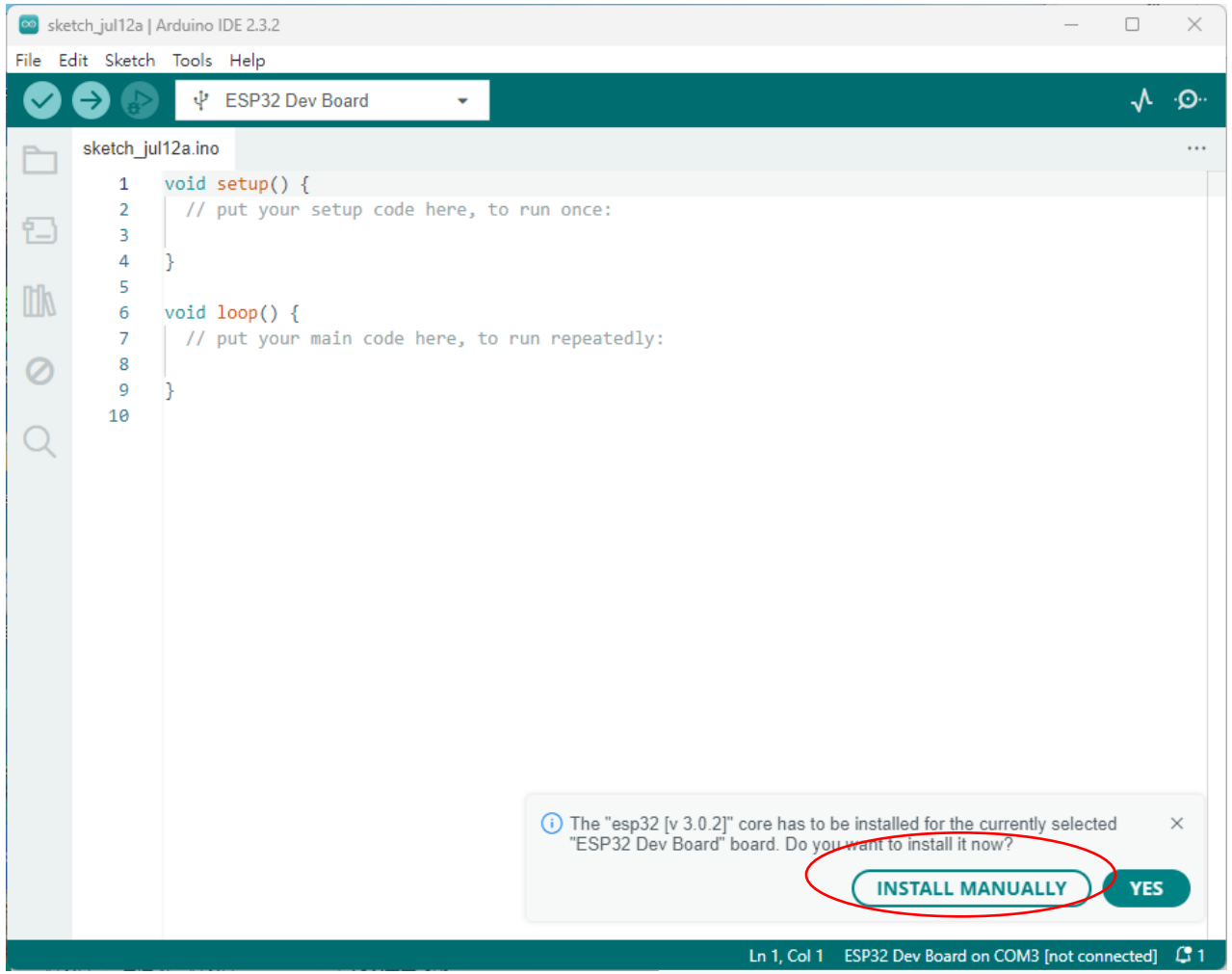


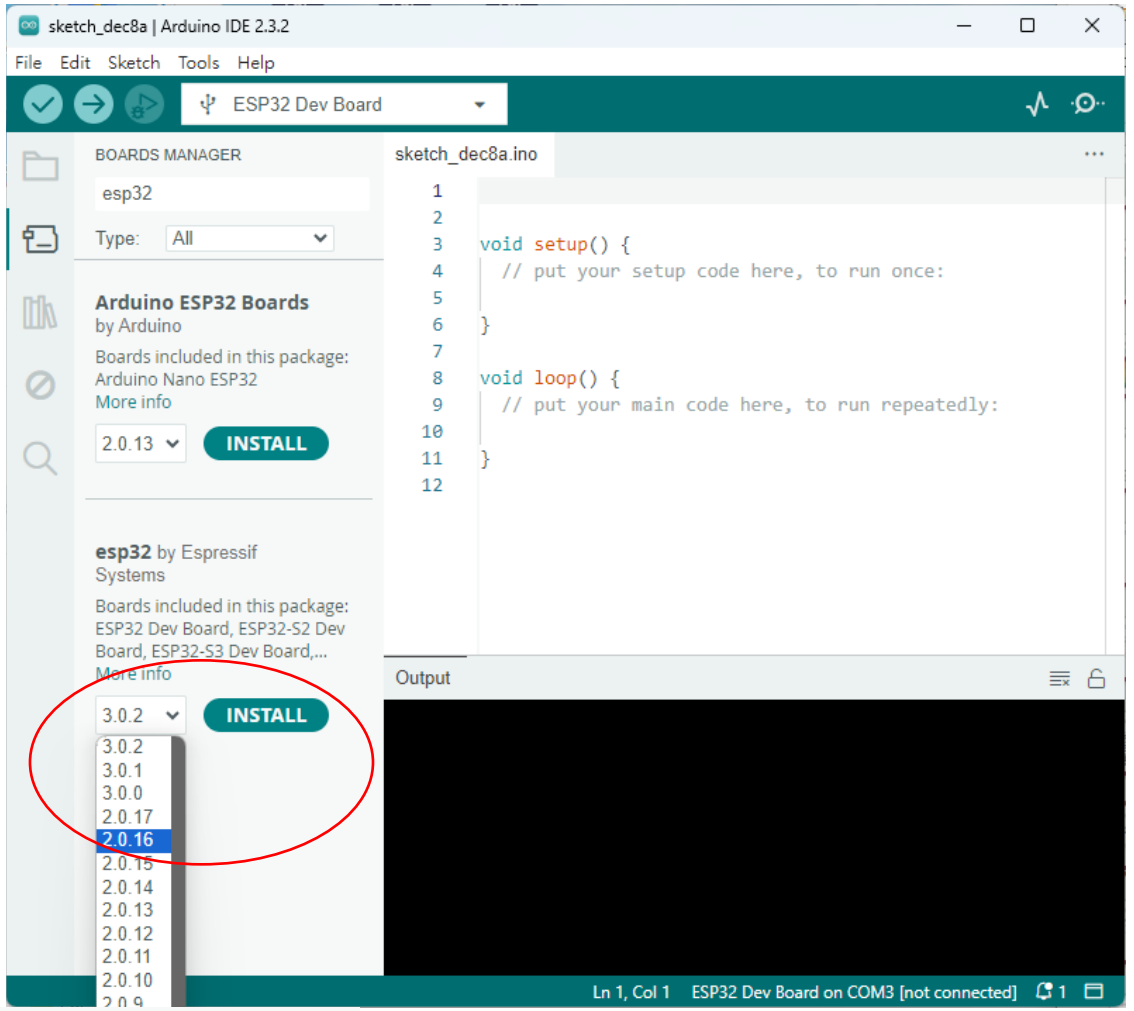
Configuration for Arduino ide

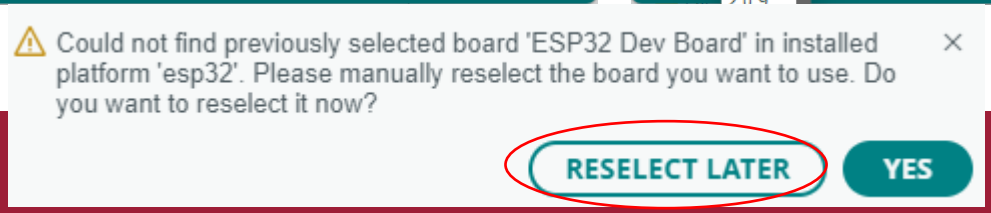
- Steps:
  3. Open Boards Manager install the ESP32 Dev Module, version 2.0.16



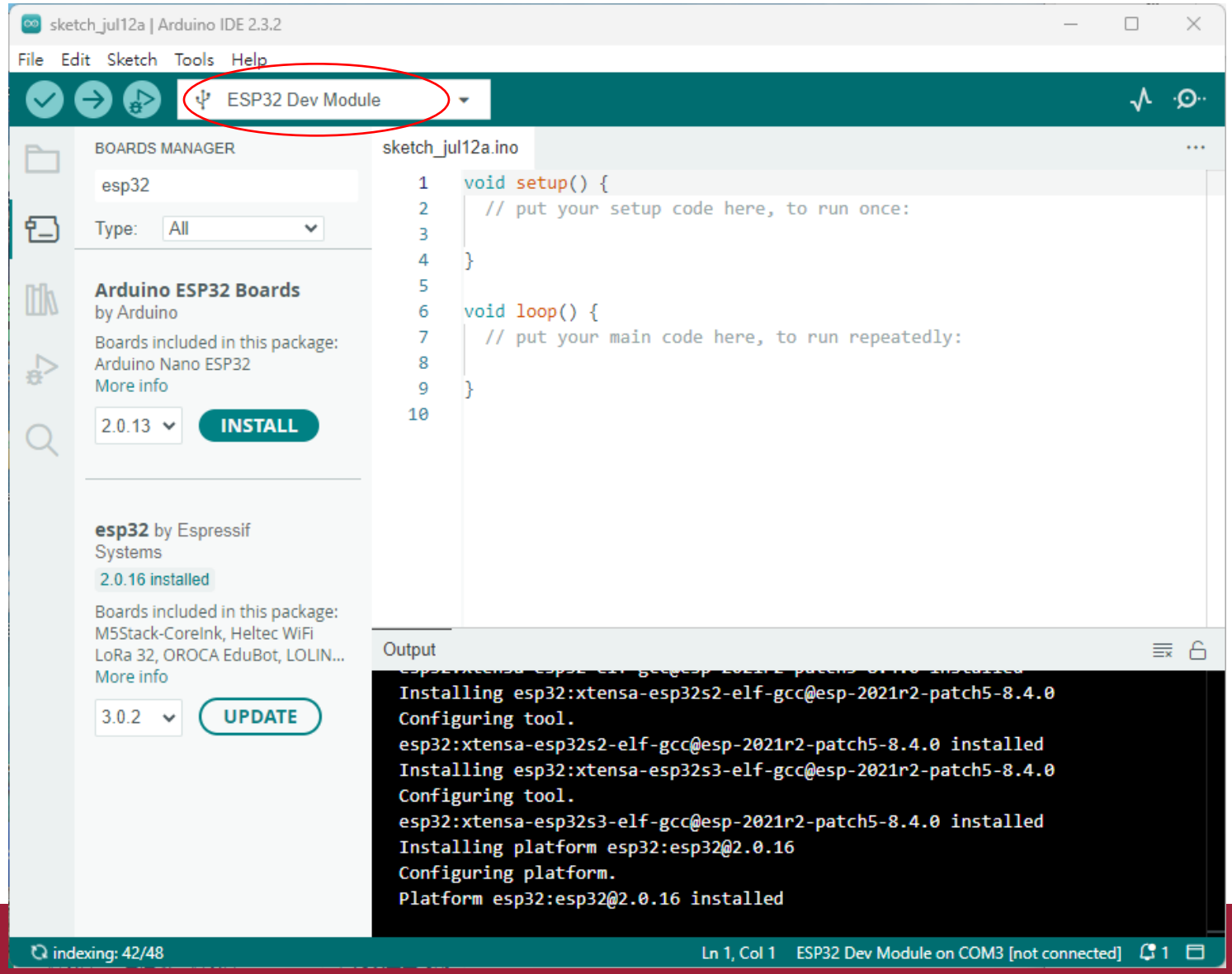
Configuration for Arduino ide





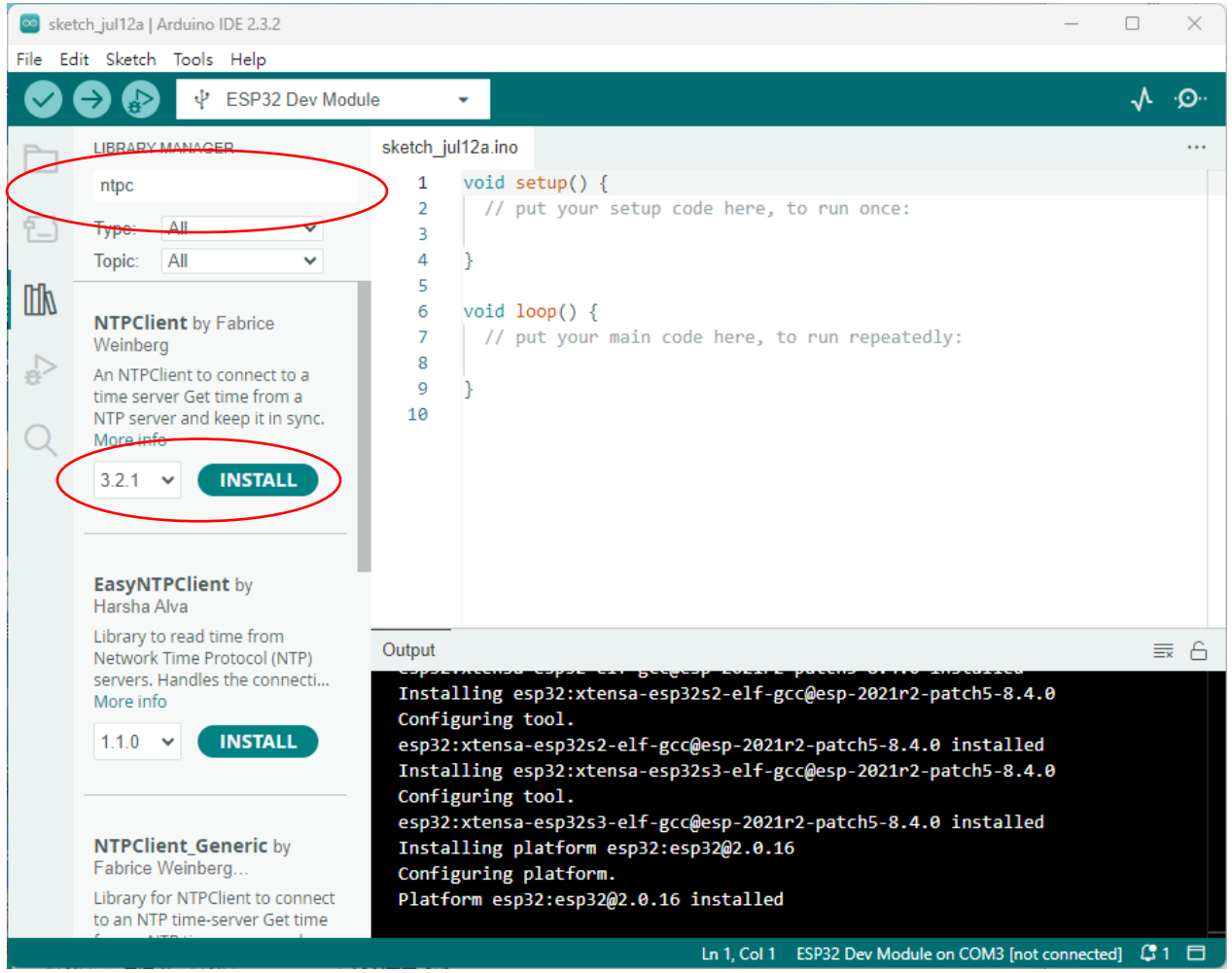


Configuration for Arduino ide



Configuration for Arduino ide

- Steps:
  4. Open Mange Libraries from the Sketch | Include Library menu and install the following libraries
     - NTPClient by Fabrice Weinberg Version 3.2.1
     - Arduinojson by Benoit Blanchon Version 6.19.4
     - ESP32Time 2.0.0 by fblego Version

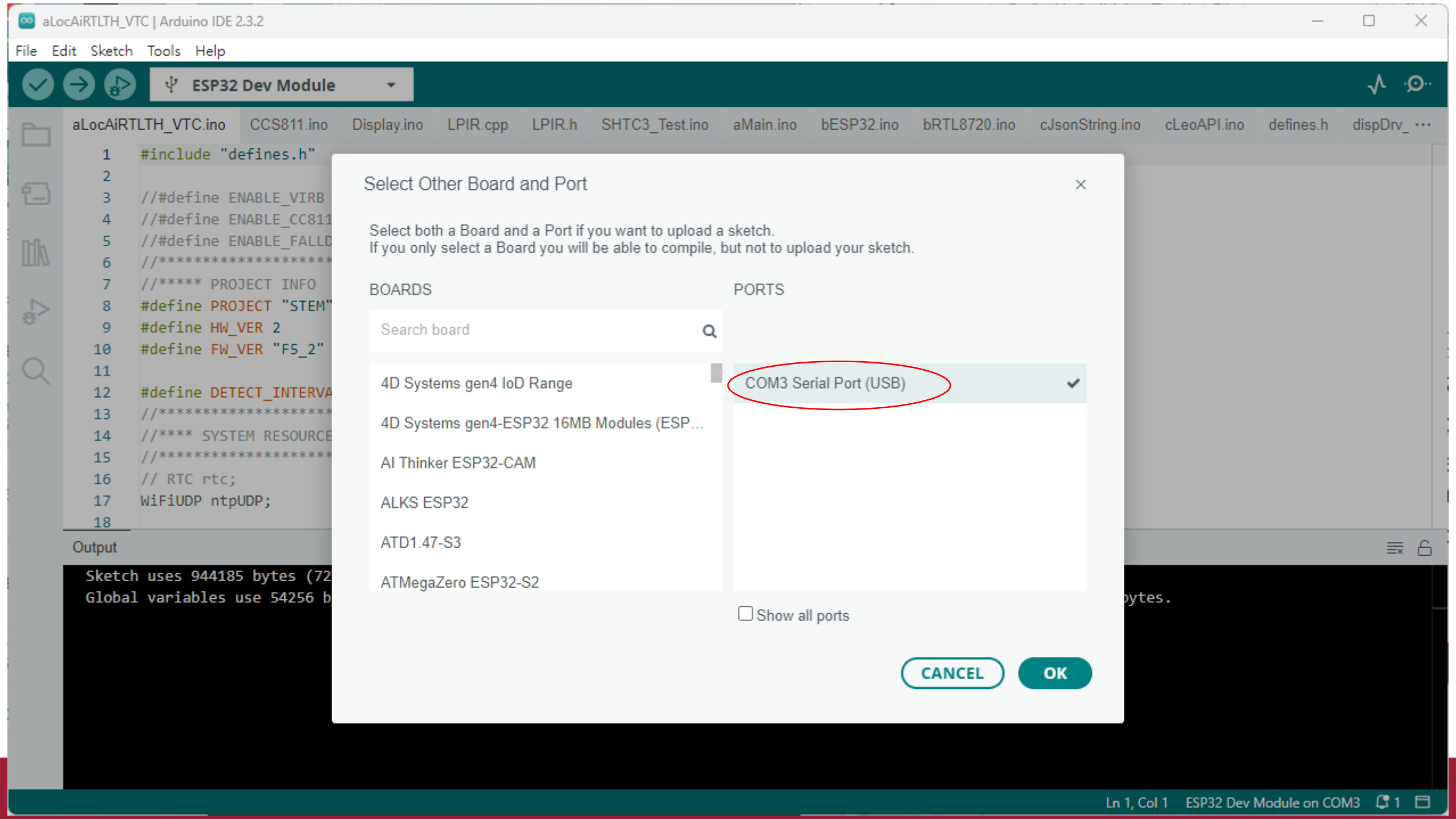


Resource management using Github

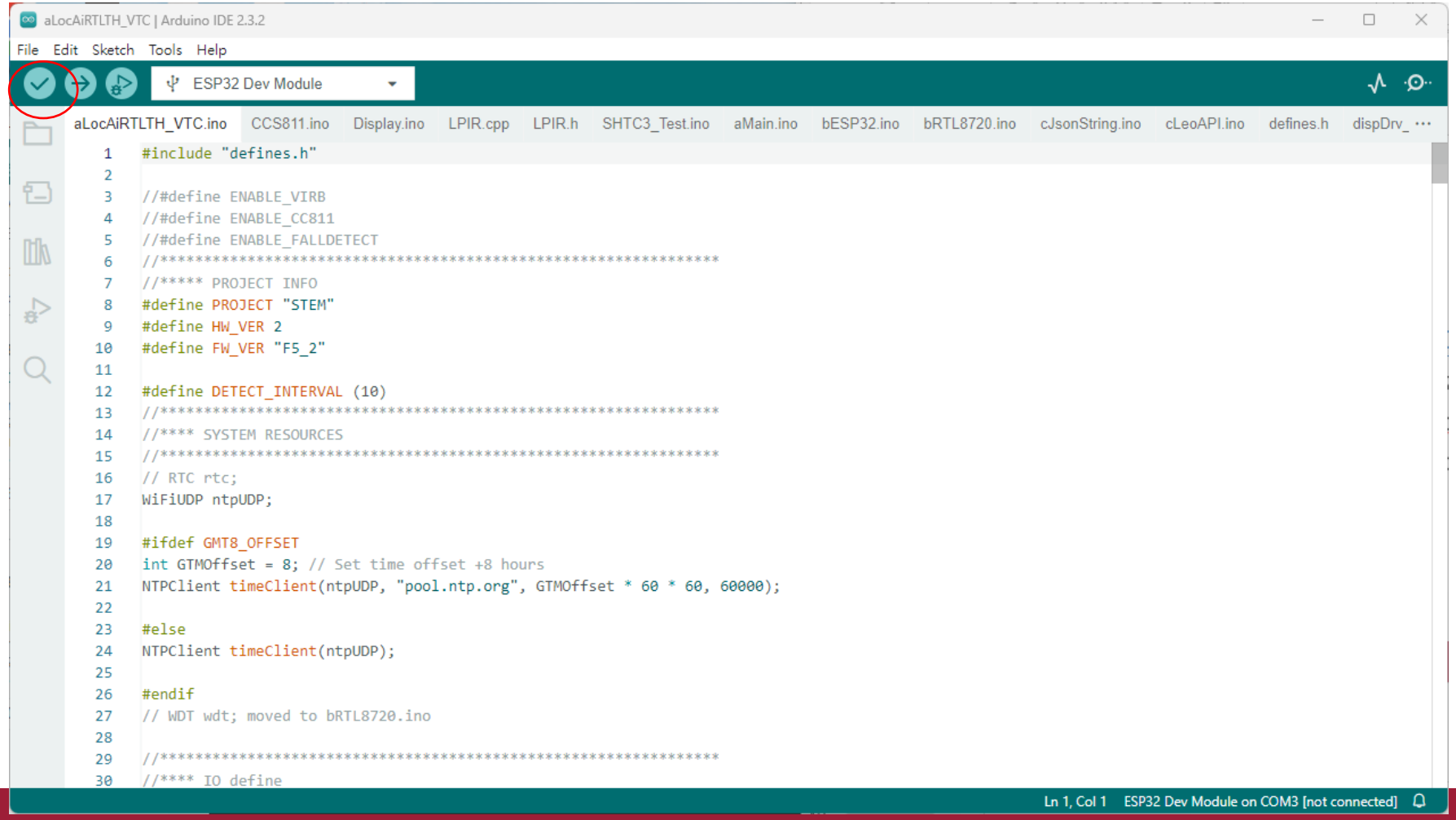
- The resources can be assessed in

https://github.com/jameschau2014/IoTplatform

- Download aLocAiRTLTH.zip, unzip it to the document->Arduino folder
- Open the aLocAiRTLTH_VTC.ino in the Arduino IDE
- Connect the STEM board to the computer via the USB cable and set the appropriate COM port in the Arduino IDE.
- Press the Verify button to compile the programs


aLocAiRTLTH_VTC | Arduino IDE 2.3.2
File Edit Sketch Tools Help
ESP32 Dev Module
aLocAiRTLTH_VTC.ino CCS811.ino Display.ino LPIR.cpp LPIR.h SHTC3_Test.ino aMain.ino bESP32.ino bRTL8720.ino cJsonString.ino cLeoAPI.ino defines.h dispDrv_
1 #include "defines.h"
2
3 //#define ENABLE_VIRB
4 //#define ENABLE_CC811
5 //#define ENABLE_FALLD
6 //**********
7 //***** PROJECT INFO
8 #define PROJECT "STEM"
9 #define HW_VER 2
10 #define FW_VER "F5_2"
11
12 #define DETECT_INTERVA
13 //**********
14 //**** SYSTEM RESOURCE
15 //**********
16 // RTC rtc;
17 WiFiUDP ntpUDP;
18
Output
Sketch uses 944185 bytes (72
Global variables use 54256 b
ytes.
Select Other Board and Port
Select both a Board and a Port if you want to upload a sketch.
If you only select a Board you will be able to compile, but not to upload your sketch.
BOARDS
PORTS
Search board
4D Systems gen4 IoD Range
4D Systems gen4-ESP32 16MB Modules (ESP...
AI Thinker ESP32-CAM
ALKS ESP32
ATD1.47-S3
ATMegaZero ESP32-S2
COM3 Serial Port (USB)
Show all ports
CANCEL
OK
Ln 1, Col 1 ESP32 Dev Module on COM3 1

Compile and upload the program



aLocAiRTLTH_VTC | Arduino IDE 2.3.2

File Edit Sketch Tools Help

ESP32 Dev Module

aLocAiRTLTH_VTC.ino | CCS811.ino | Display.ino | LPIR.cpp | LPIR.h | SHTC3_Test.ino | aMain.ino | bESP32.ino | bRTL8720.ino | cJsonString.ino | cLeoAPI.ino | defines.h | dispDrv_

```
1   #include "defines.h"
2
3   //#define ENABLE_VIRB
4   //#define ENABLE_CC811
5   //#define ENABLE_FALLDETECT
6   //*****************************************************************
7   //***** PROJECT INFO
8   #define PROJECT "STEM"
9   #define HW_VER 2
10  #define FW_VER "F5_2"
11
12  #define DETECT_INTERVAL (10)
13  //*****************************************************************
14  //**** SYSTEM RESOURCES
15  //*****************************************************************
16  // RTC rtc;
17  WiFiUDP ntpUDP;
18
```

Output

```
Writing at 0x000e5d41... (92 %)
Writing at 0x000eb780... (94 %)
Writing at 0x000f0971... (97 %)
Writing at 0x000f62ab... (100 %)
Wrote 950768 bytes (611644 compressed) at 0x00010000 in 9.8 seconds (effective 777.3 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
```

Success!

Ln 1, Col 1 ESP32 Dev Module on COM3 2

WiFi Connection

- In the source code aLocAiRTLTH_VTC.ino, find line 444 and 445, change your WiFi access SSID and password

```
436  void app_setup_init_wiFiList(void)
437  {
438    strcpy(apList[0].AP_SSID, "locationap");
439    strcpy(apList[0].AP_PWD, "locationai321");
440    strcpy(apList[1].AP_SSID, "Mapxus");
441    strcpy(apList[1].AP_PWD, "5mart&5imple");
442    strcpy(apList[2].AP_SSID, "update@TrackingDevice"); // special AP to denote OTA update
443    strcpy(apList[2].AP_PWD, "1234567890");
444    strcpy(apList[3].AP_SSID, "dummy_ap");
445    strcpy(apList[3].AP_PWD, "dummy_pw");
```

Sensor connection and result display

- Connect CO2 or TEMP sensor to any port on the board via the cable
- Reset the STEM board by pressing the red button
- Open the Serial Monitor in the Tools Menu and set the baud rate to 115200
- The result can be displayed on the Serial Monitor
- Notice that the Air quality will be displayed on the onboard OLED display

aLocAiRTLTH_VTC | Arduino IDE 2.3.2

File Edit Sketch Tools Help

ESP32 Dev Module

aLocAiRTLTH_VTC.ino | CCS811.ino | Display.ino | LPIR.cpp | LPIR.h | SHTC3_Test.ino | aMain.ino | bESP32.ino | bRTL8720.ino | cJsonString.ino | cLeoAPI.ino | defines.h | dispDrv_ ...

```
80
81  //*****************************************************************
82  //*****************************************************************
83  void setup()
84  {
85    // Initialize serial and wait for port to open:
86
87    Serial.begin(115200);
88    while (!Serial)
89    {
90      ; // wait for serial port to connect. Needed for native USB port only
91    }
92
93    // // check for the presence of the shield:
94    // if (WiFi.status() == WL_NO_SHIELD) {
95    //     Serial.println("WiFi shield not present");
96    //     // don't continue:
97    //     while (true);
```

Output | Serial Monitor ×

Message (Enter to send message to 'ESP32 Dev Module' on 'COM3')

New Line | 115200 baud

```
Humidity:0%  Temperature:0C  PIR: 1  LastTime: 05:24:10
app_CC811_loop: 8400
eCo2[8]: 400, TVOC[8]: 0, Selected: 4, RAW: 526
Humidity:0%  Temperature:0C  PIR: 1  LastTime: 05:24:11
app_CC811_loop: 9607
eCo2[9]: 607, TVOC[9]: 31, Selected: 4, RAW: 477
Avg humi: 0, Avg temp: 0
Avg eCo2: 464, Avg TVOC: 9
```

Ln 445, Col 37 ESP32 Dev Module on COM3 2

Test the data logging

- Modify the code in line 102 of SHTC3_Test.ino, “client.print(String("GET /dweet/for/myesp8266?temperature=")…” into “client.print(String("GET /dweet/for/your_name?temperature=")”

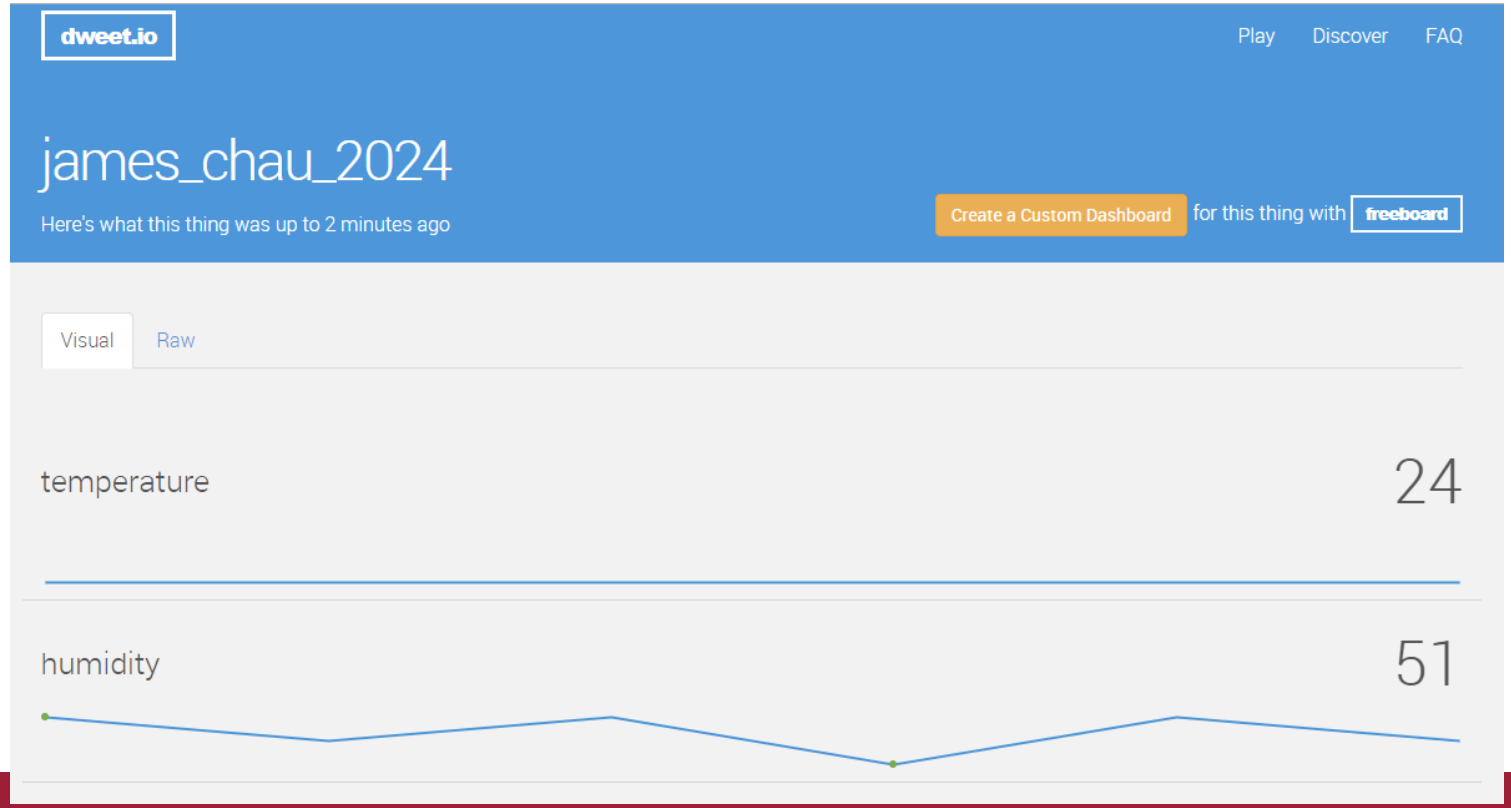
```
96      if (SHTC3_CRC_CHECK(RH_temp, RH_CRC))
97      {
98        RH = float(RH_temp) * 100 / 65536;
99        SHTC3_data.humi[sys.readCnt] = RH;
100     }
101   }
102       client.print(String("GET /dweet/for/james_chau_2024?temperature=") + String(T) + "&humidity="
103       delay(10);
104       while(client.available()){
105       String line = client.readStringUntil('\r');
106         Serial.print(line);
107       }
108 }
```

Test the data logging

- Read your logged data by assessing: https://dweet.me:3334/get/latest/yoink/from/your_name

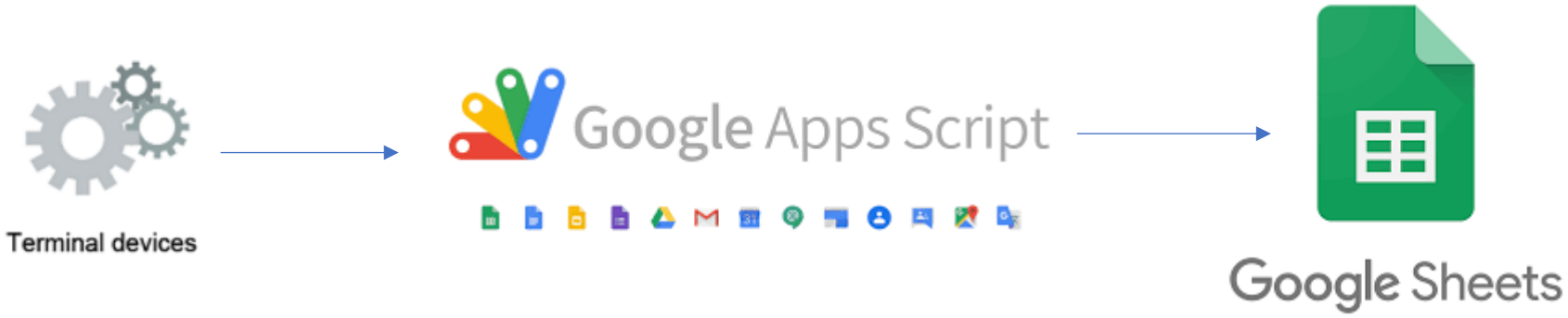
Show data in dash board

- Access: https://dweet.io/follow/james_chau_2024



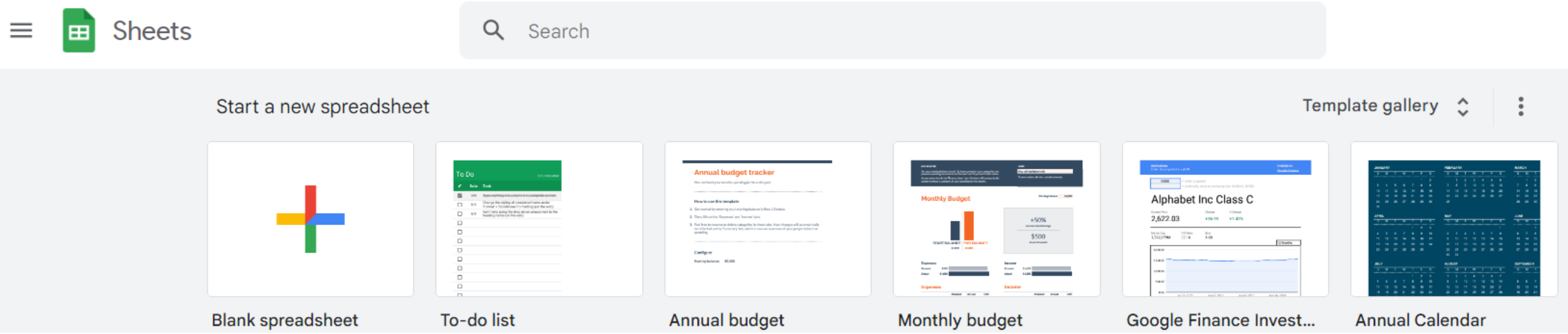
Data logging using Google sheet and apps script

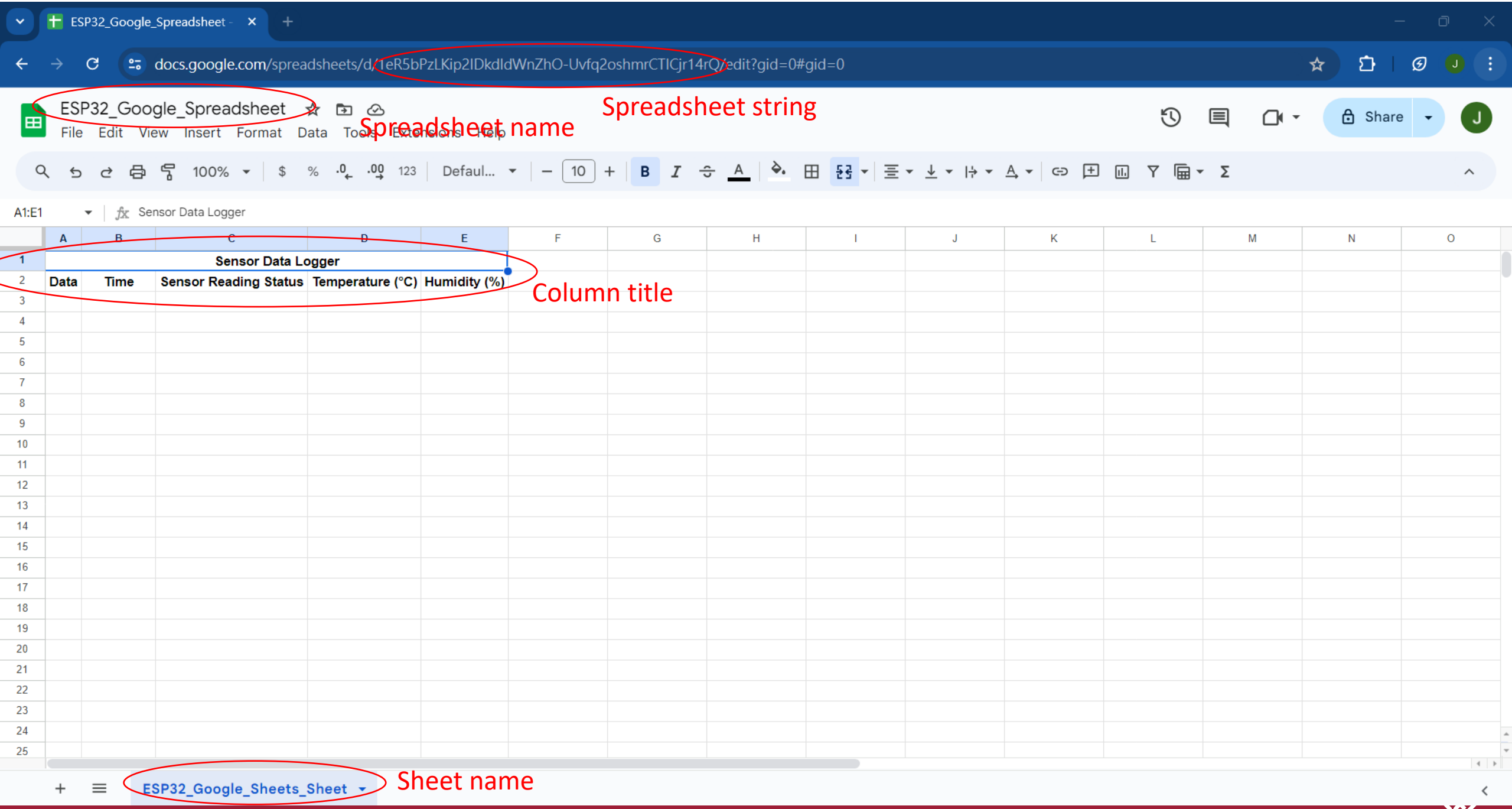
- Data collected by the IoT device can be logged into an online spreadsheet for further analysis
  - Google sheets and app script script



Prepare a Google Sheet

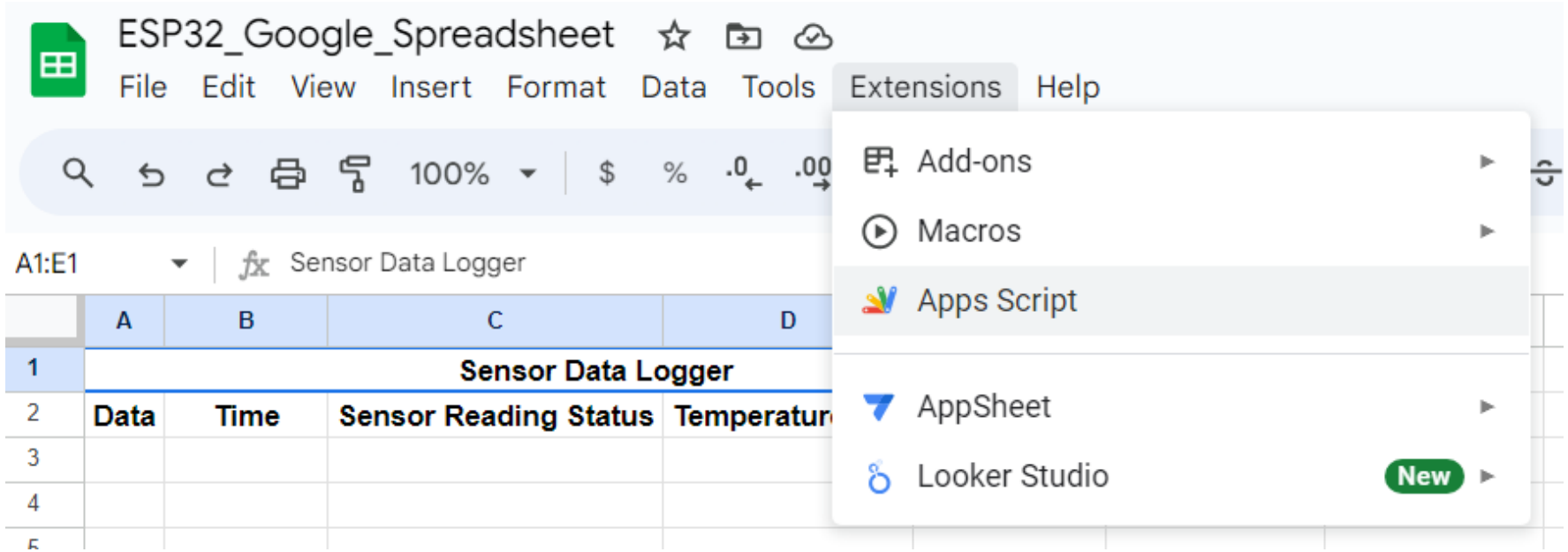
- Click blank spreadsheet in Google Sheet
- And input the necessary data shown on new page




ESP32_Google_Spreadsheet
docs.google.com/spreadsheets/d/1eR5bPzLKip2IDkdIdWnZhO-Uvfq2oshmrCTICjr14rQ/edit?gid=0#gid=0
ESP32_Google_Spreadsheet
Spreadsheet name
Spreadsheet string
File Edit View Insert Format Data Tools Extensions Help
Share
A1:E1
Sensor Data Logger
Sensor Data Logger
Data
Time
Sensor Reading Status
Temperature (°C)
Humidity (%)
Column title
ESP32_Google_Sheets_Sheet
Sheet name

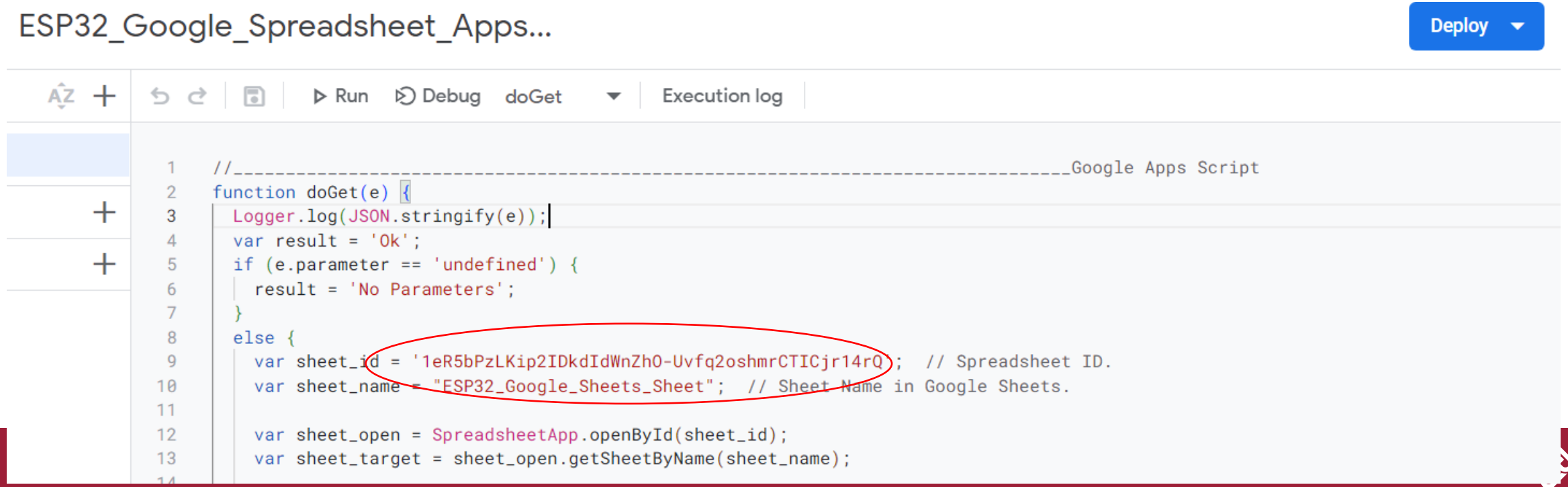
Open the Apps Script

- Press Apps Script in the Extensions menu



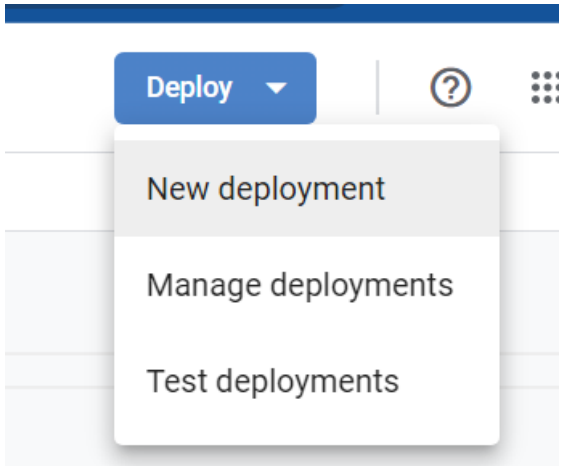
Modify the Apps Script

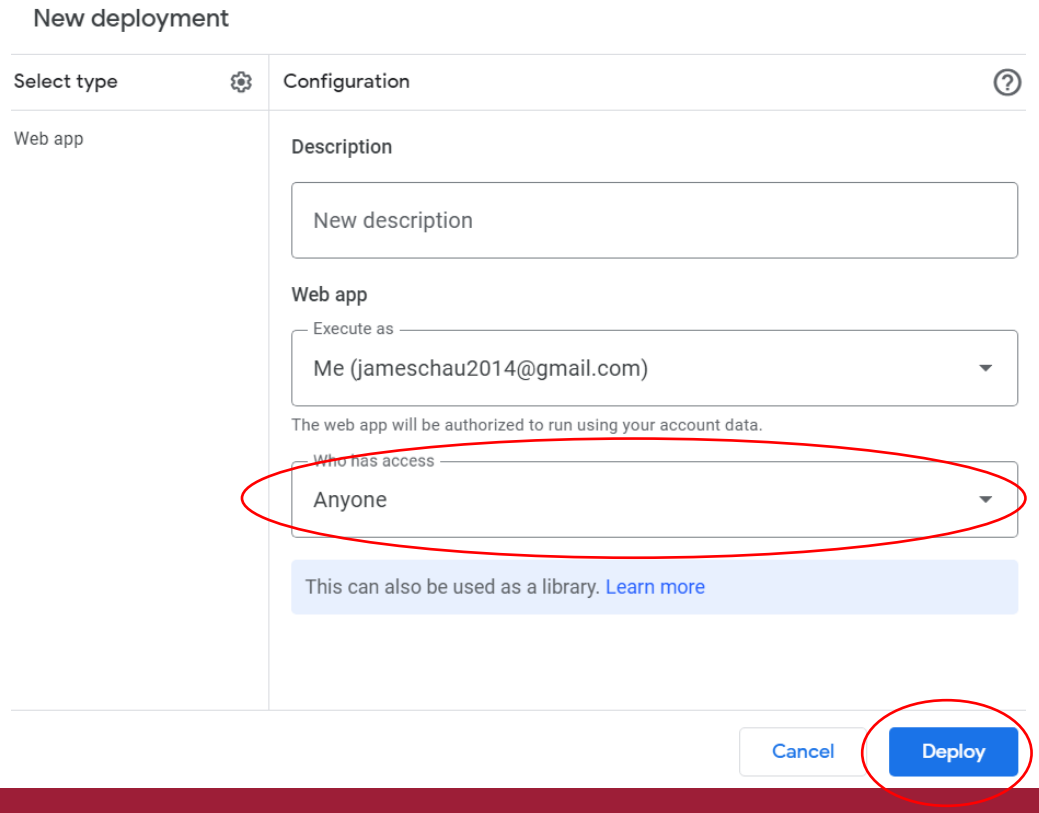
- Copy the program in the apps script template.txt into the apps script editor
- Change the sheet_id and sheet_name in lines 9 and 10 with your own Spreadsheet string and Sheet name



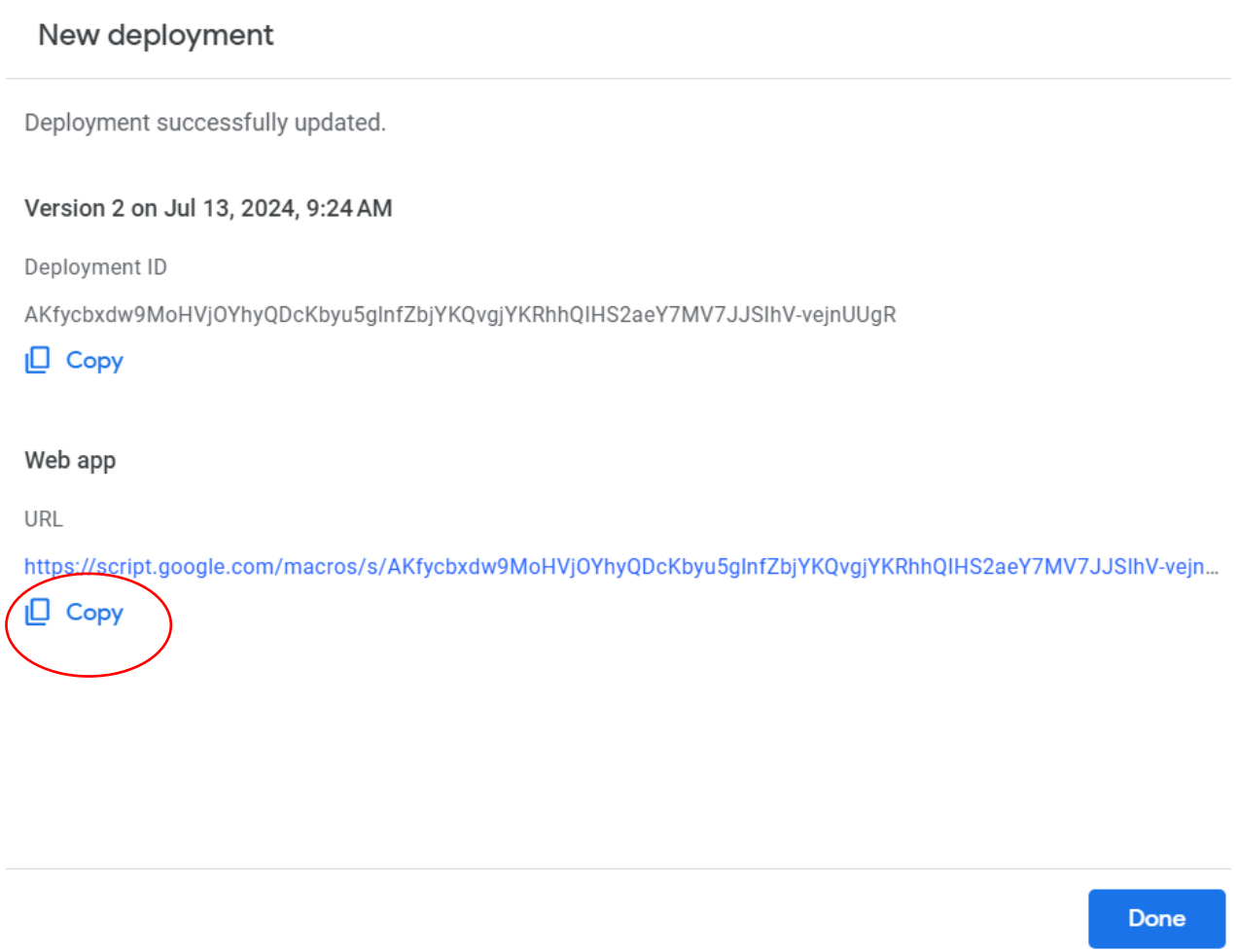
Deploy the Apps Script

- Press the deploy button->new deployment
- Change Anyone to “who has access”, then press deploy





Copy the web app URL

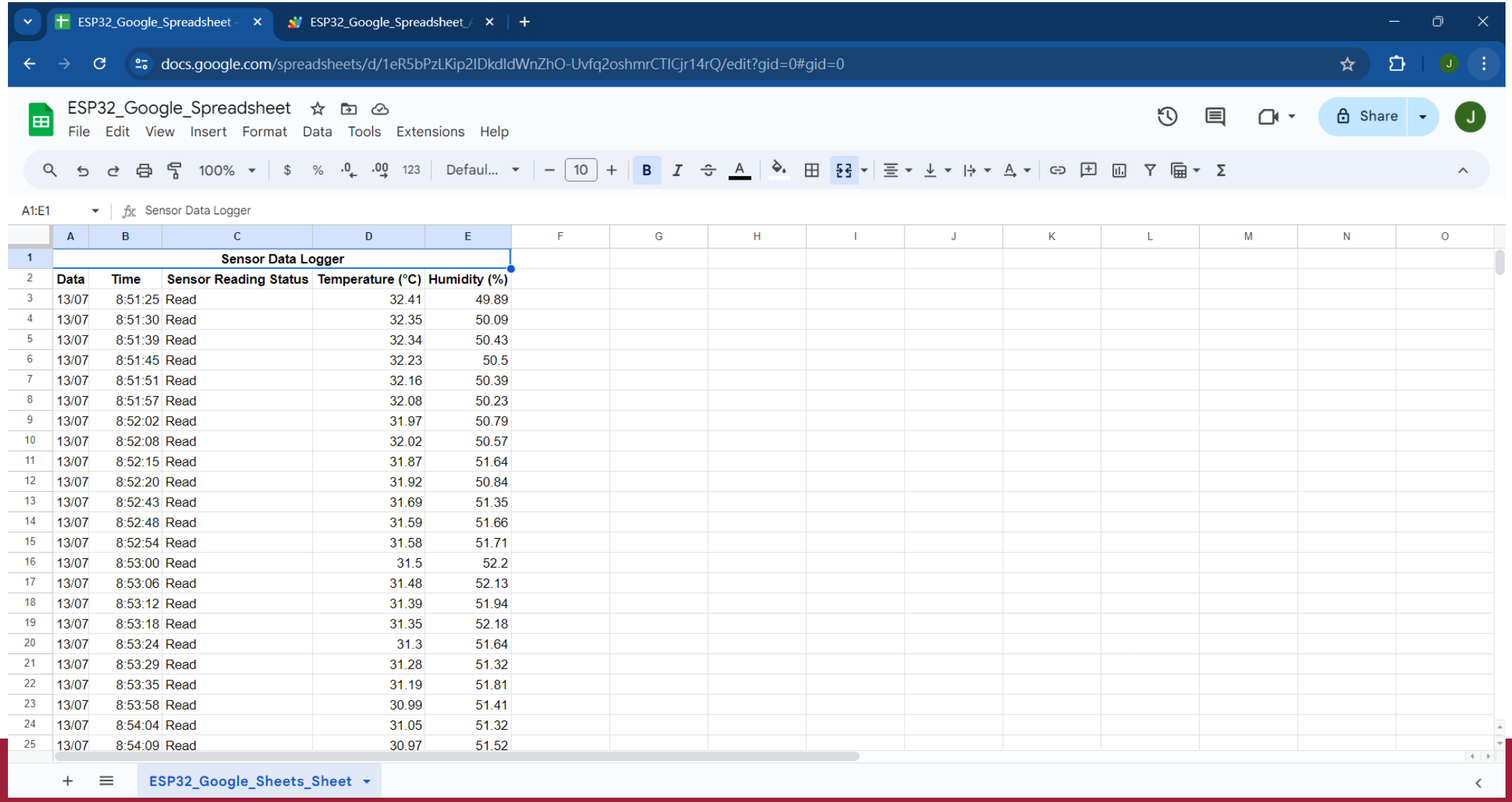


Modify the program

- In the source code SHTC3_Test.ino, find line 7 and paste your web app URL string
- Upload the program to the STEM board

```
1  #define SHTC3_VERSION "SHTC3-2"
2  #define SHTC3_ADDRESS 0x70
3  float T, RH;
4  const char* host = "dweet.io";
5
6  // Google script Web_App_URL.
7  String Web_App_URL = "https://script.google.com/macros/s/AKfycbxdw9MoHVjOYhyQDcKbyu5glnfZbjYKQvgjYKRhhQIHS2aeY7MV7JJSIhV-vejnUUgR/exec";
8  //change your web app url
9
```

Logging results



	A	B	C	D	E
1	Sensor Data Logger				
2	Data	Time	Sensor Reading Status	Temperature (°C)	Humidity (%)
3	13/07	8:51:25	Read	32.41	49.89
4	13/07	8:51:30	Read	32.35	50.09
5	13/07	8:51:39	Read	32.34	50.43
6	13/07	8:51:45	Read	32.23	50.5
7	13/07	8:51:51	Read	32.16	50.39
8	13/07	8:51:57	Read	32.08	50.23
9	13/07	8:52:02	Read	31.97	50.79
10	13/07	8:52:08	Read	32.02	50.57
11	13/07	8:52:15	Read	31.87	51.64
12	13/07	8:52:20	Read	31.92	50.84
13	13/07	8:52:43	Read	31.69	51.35
14	13/07	8:52:48	Read	31.59	51.66
15	13/07	8:52:54	Read	31.58	51.71
16	13/07	8:53:00	Read	31.5	52.2
17	13/07	8:53:06	Read	31.48	52.13
18	13/07	8:53:12	Read	31.39	51.94
19	13/07	8:53:18	Read	31.35	52.18
20	13/07	8:53:24	Read	31.3	51.64
21	13/07	8:53:29	Read	31.28	51.32
22	13/07	8:53:35	Read	31.19	51.81
23	13/07	8:53:58	Read	30.99	51.41
24	13/07	8:54:04	Read	31.05	51.32
25	13/07	8:54:09	Read	30.97	51.52

Summary

- Smart Location IoT kit and IOT history are introduced
- Wireless communication technology is introduced
- Some electronic components are introduced
- Data manipulation and analysis using STEM board

Thank you…

Take away