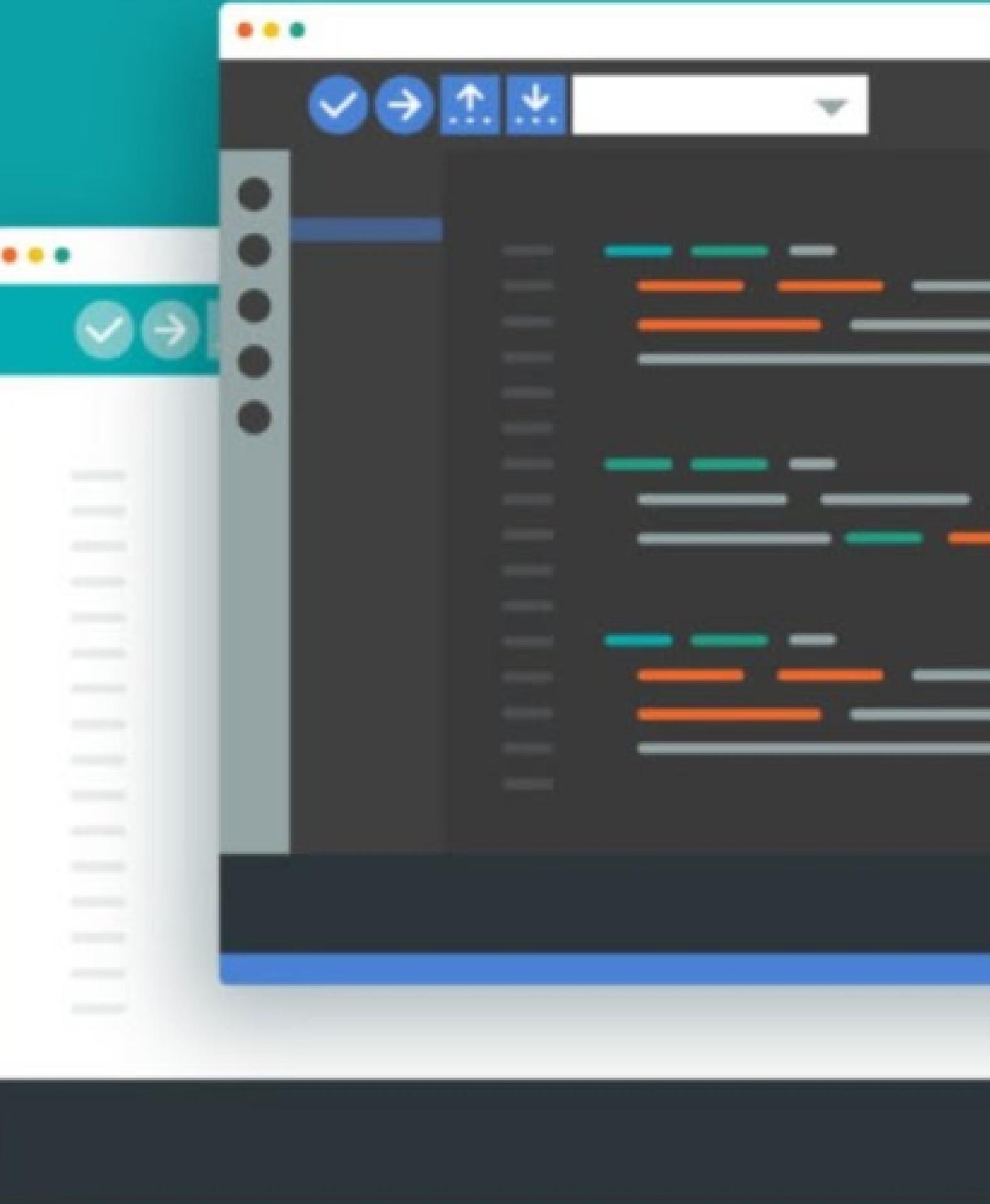


INTERNET OF THINGS

DAY 1



INSTALL

ติดตั้ง ARDUINO IDE



Arduino IDE 2.1.0

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE

The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

DOWNLOAD OPTIONS

Windows Win 10 and newer, 64 bits

Windows MSI installer

Windows ZIP file

Linux AppImage 64 bits (X86-64)

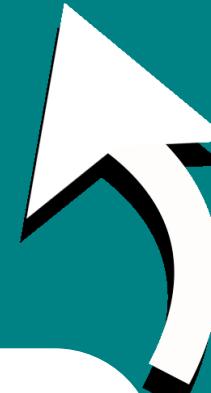
Linux ZIP file 64 bits (X86-64)

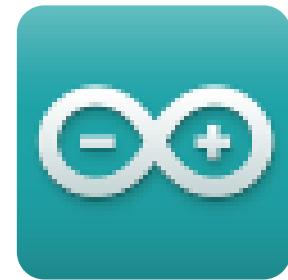
macOS Intel, 10.14: "Mojave" or newer, 64 bits

macOS Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

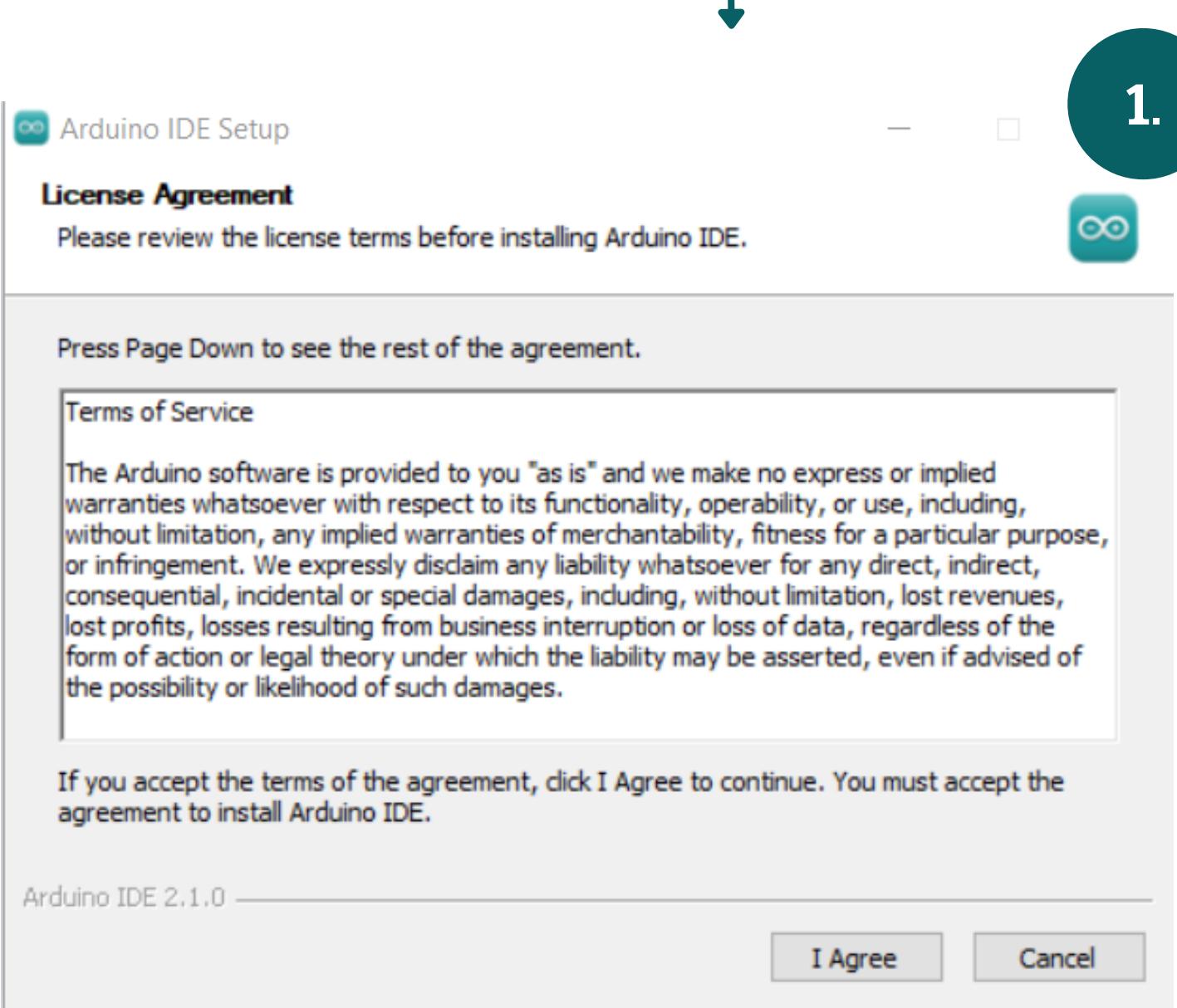
กดติดต่อเรา



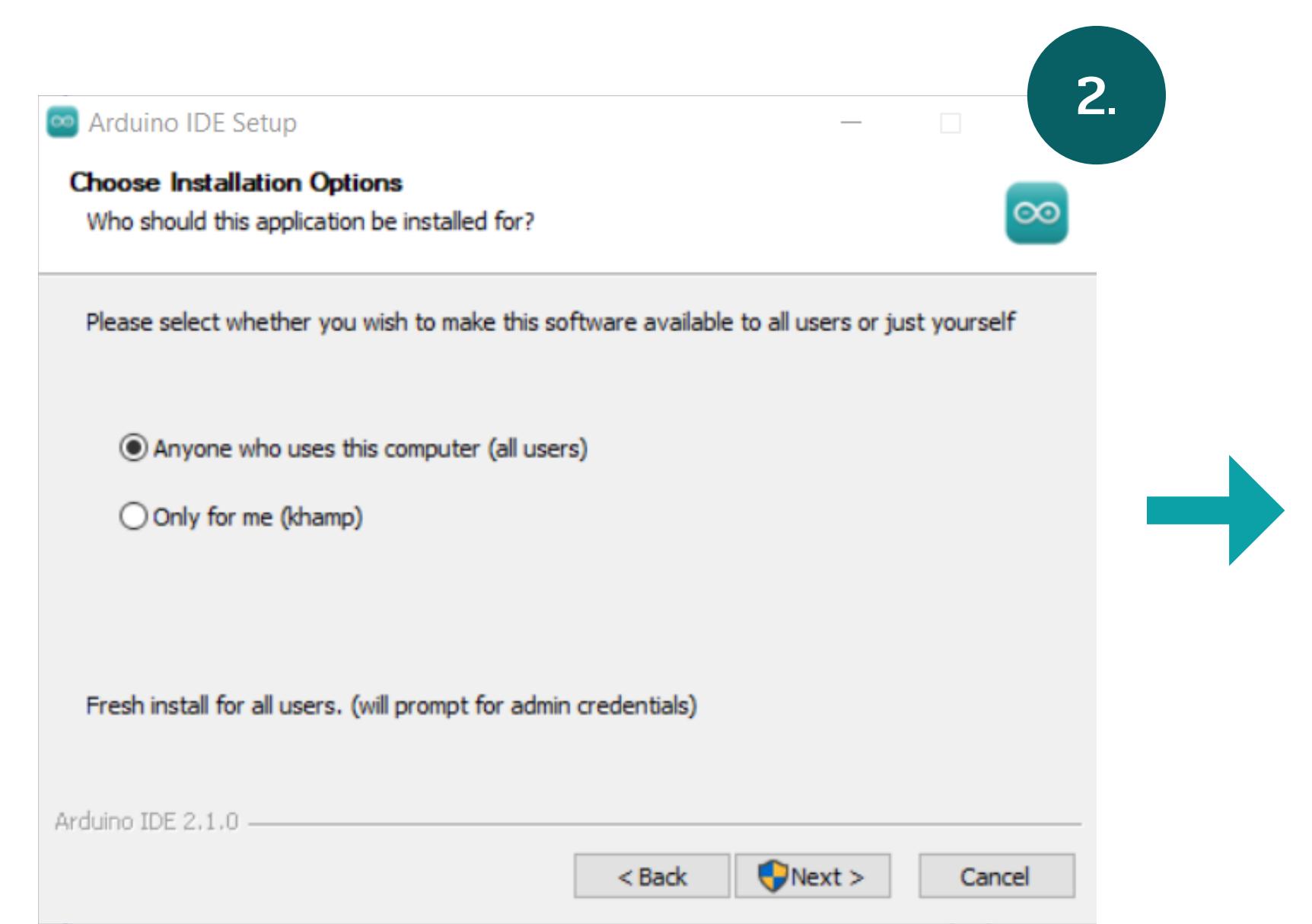


arduino-ide_2.1.0_
Windows_64bit.exe

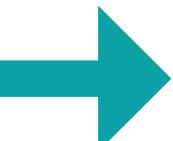
Click!

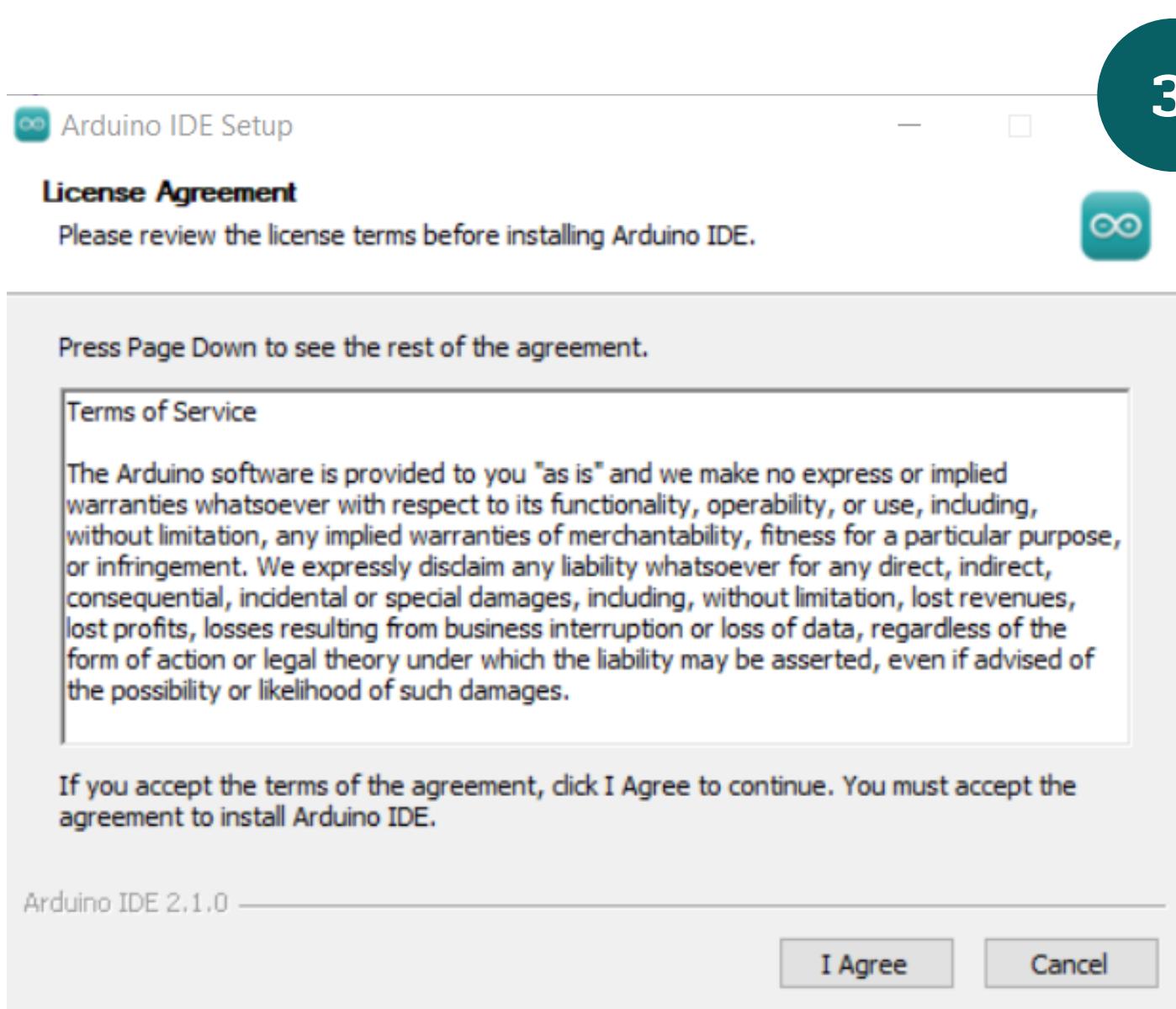


1.

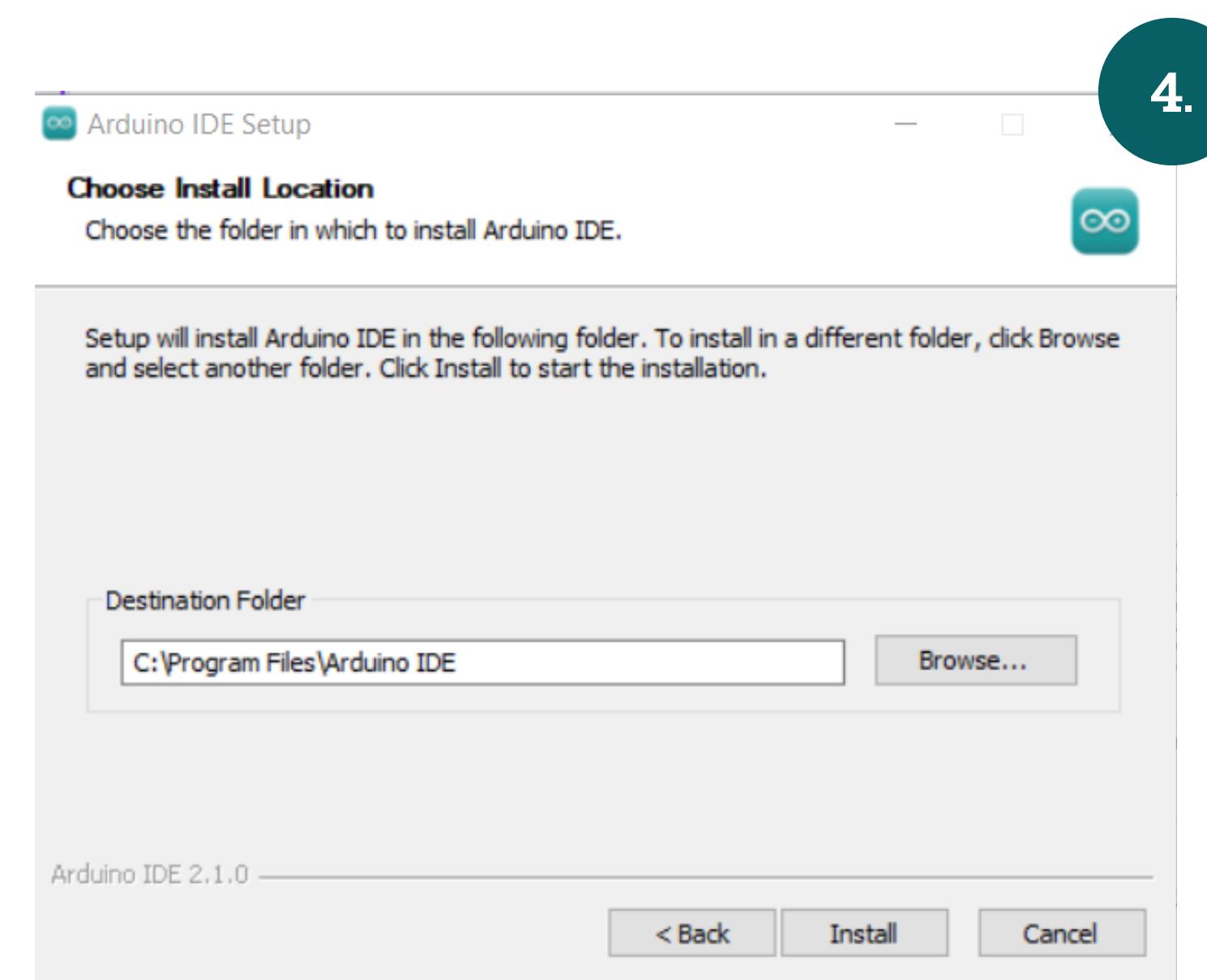


2.

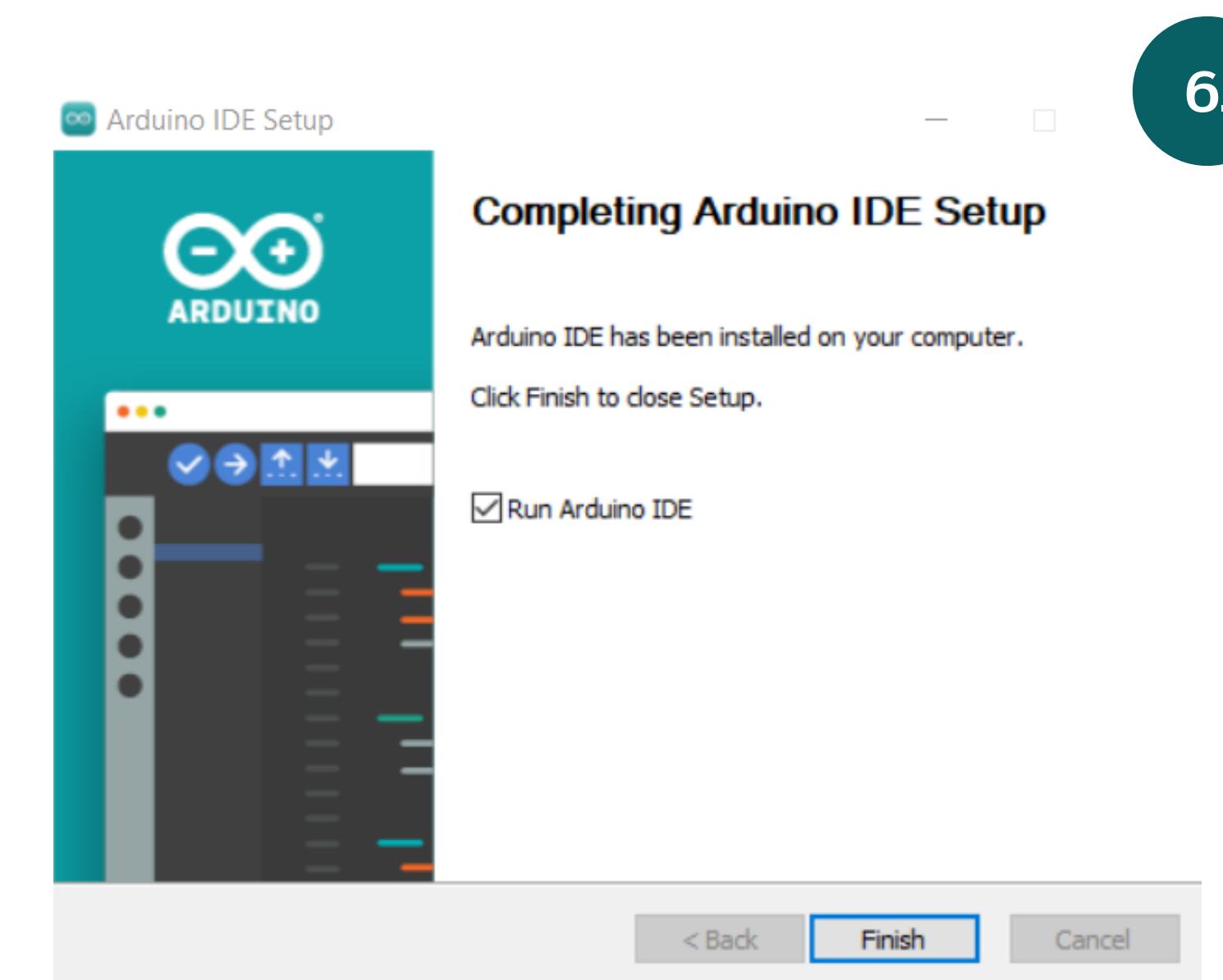
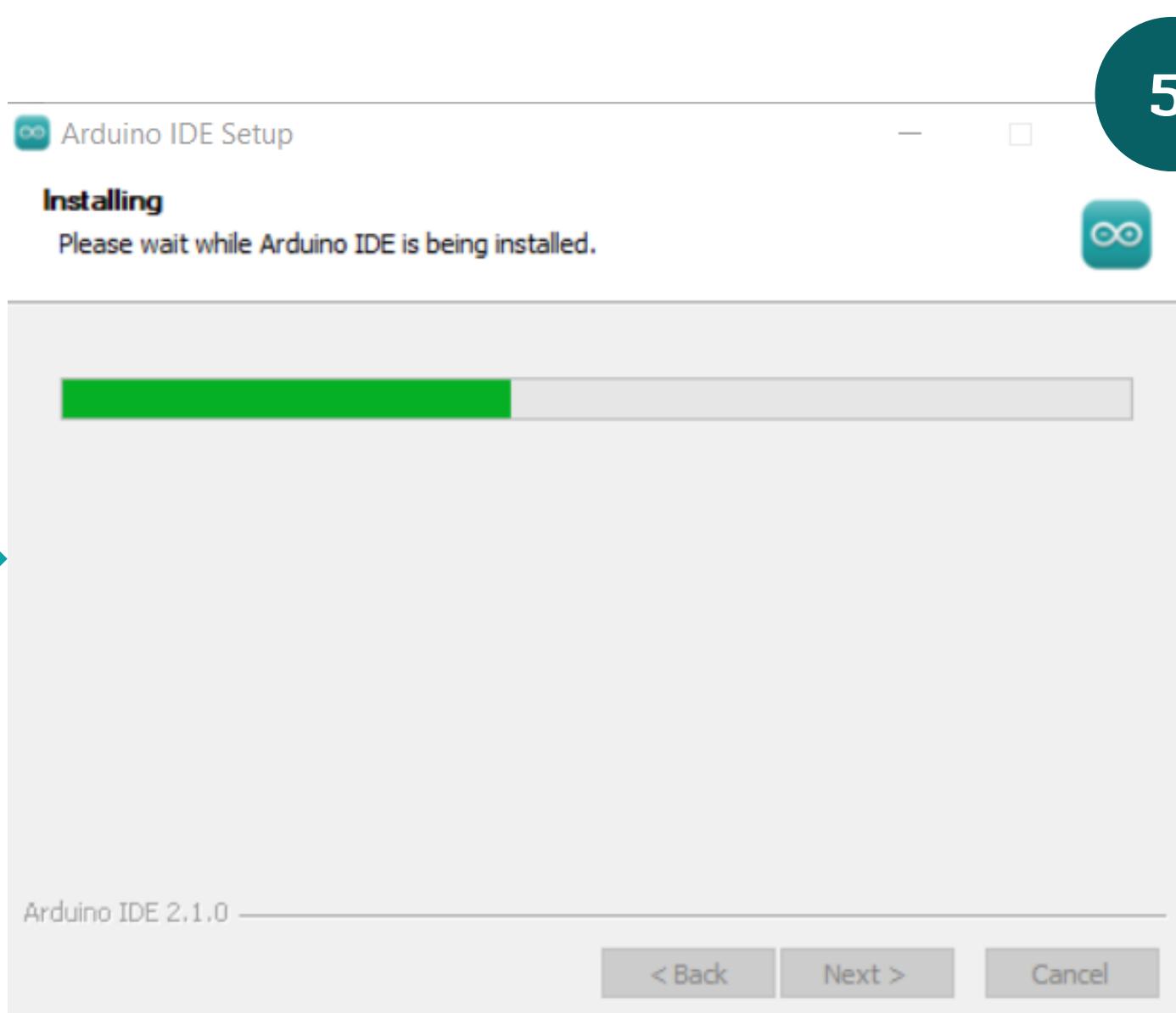




3.



4.



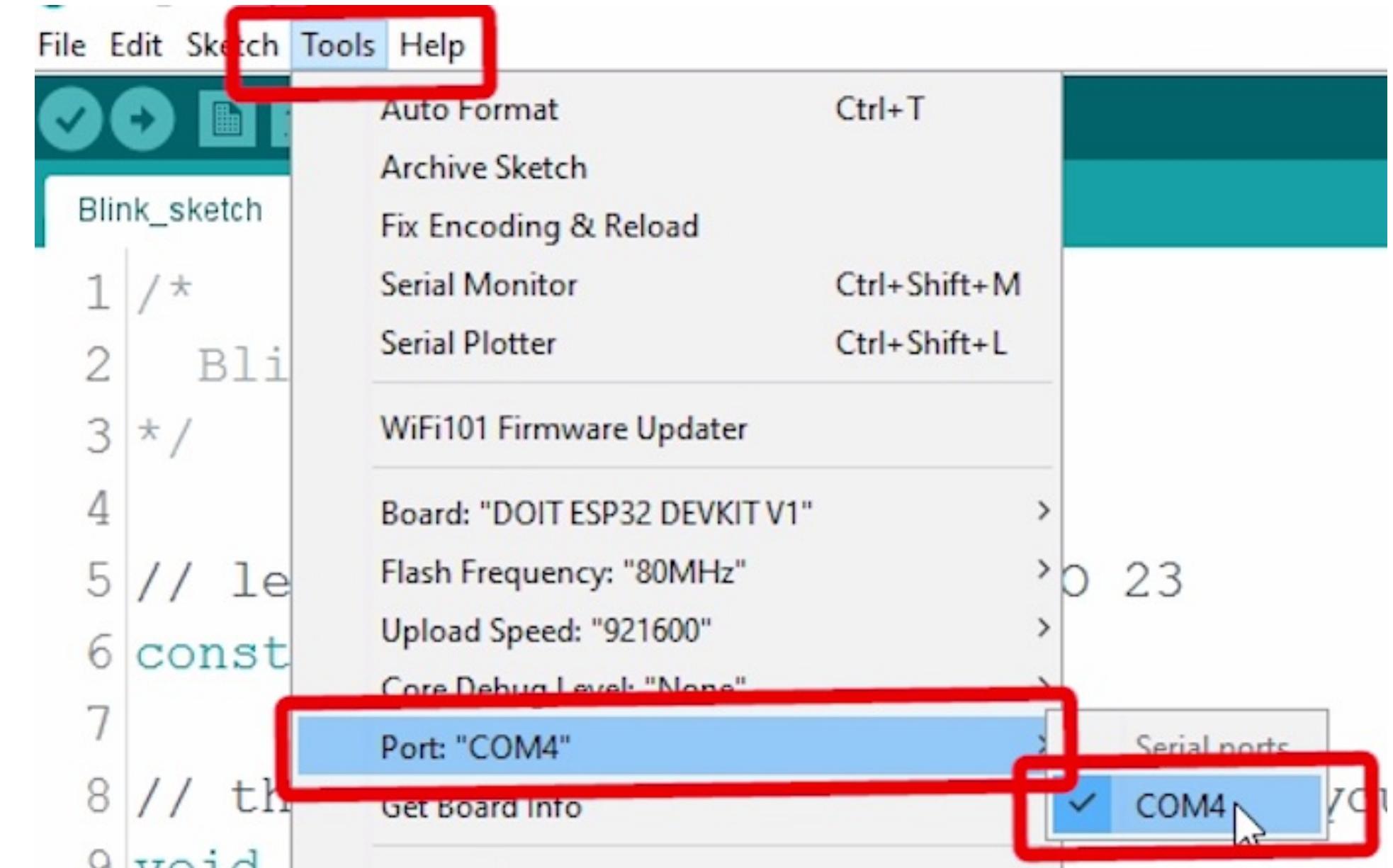
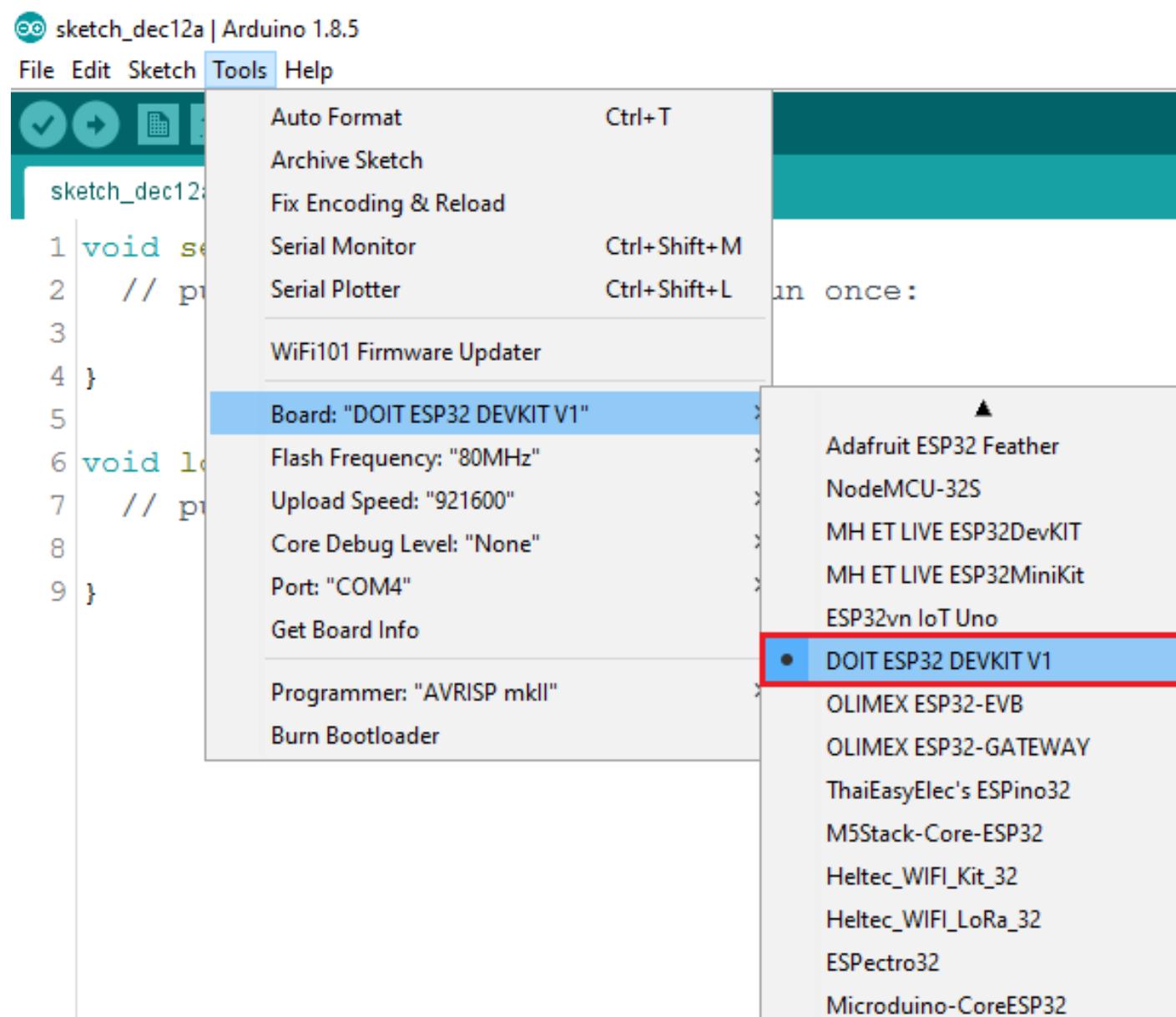
ເນື້ອໄກາດ

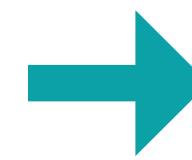
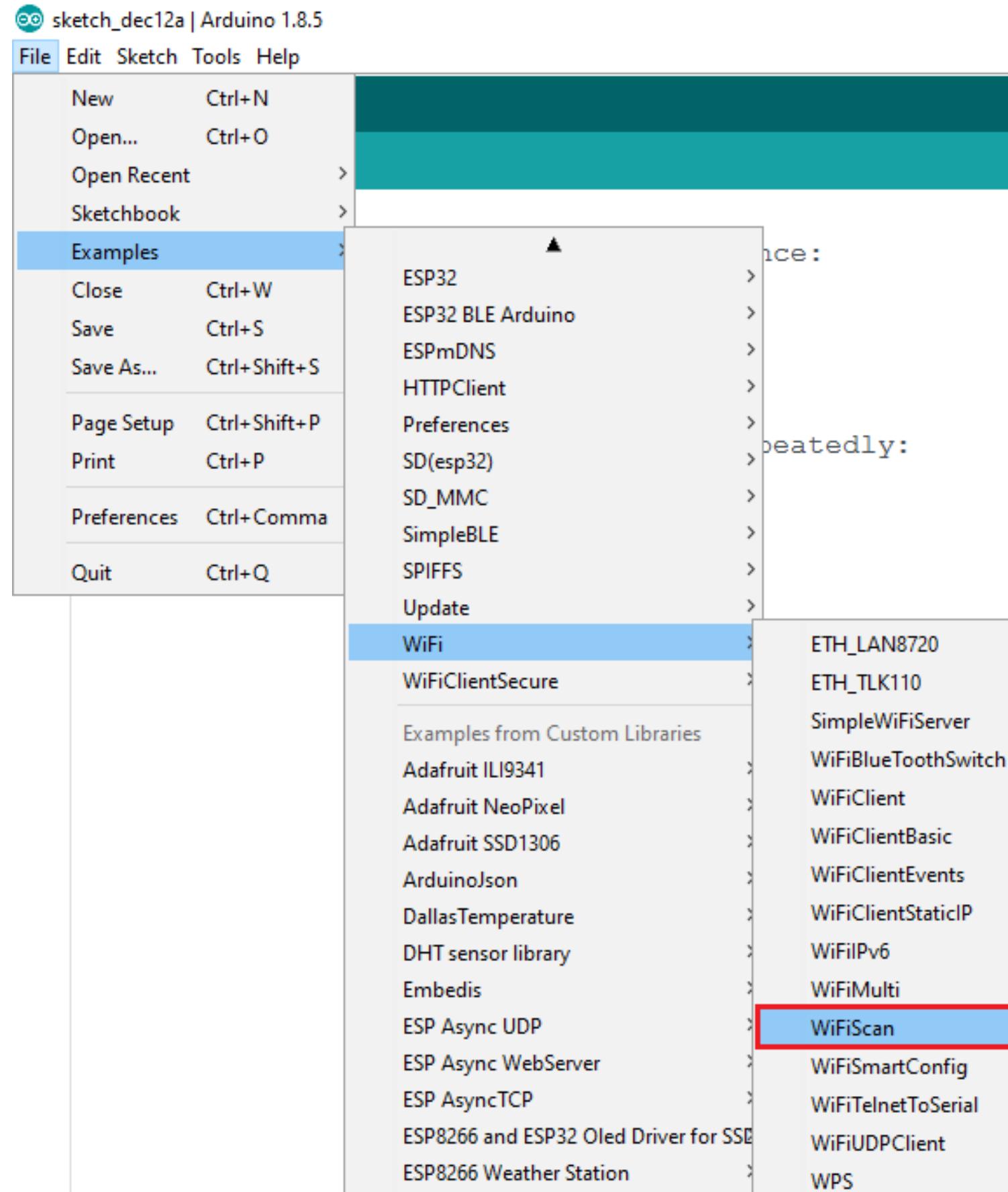


Arduino IDE

ເສີ່ອແລ້ວ!

Setting Tools





Upload!

The screenshot shows the Arduino IDE with the title bar "WiFiScan | Arduino 1.8.5". The "File" menu is open, and the upload button in the toolbar is highlighted with a red box. The main area displays the WiFiScan sketch code.

```
/*
 * This sketch demonstrates how to scan WiFi networks.
 * The API is almost the same as with the WiFi Shield library,
 * the most obvious difference being the different file you need to include:
 */
#include "WiFi.h"

void setup()
{
    Serial.begin(115200);

    // Set WiFi to station mode and disconnect from an AP if it was previously connected
    WiFi.mode(WIFI_STA);
    WiFi.disconnect();
    delay(100);

    Serial.println("Setup done");
}

void loop()
```

Examples

```
Done uploading.  
writing at 0x00004C000... (84 %)  
Writing at 0x00050000... (89 %)  
Writing at 0x00054000... (94 %)  
Writing at 0x00058000... (100 %)  
Wrote 481440 bytes (299651 compressed) at 0x00010000 in 4.7 seconds  
Hash of data verified.  
Compressed 3072 bytes to 122...  
  
writing at 0x00008000... (100 %)  
Wrote 3072 bytes (122 compressed) at 0x00008000 in 0.0 seconds (etc)  
Hash of data verified.  
  
Leaving...  
Hard resetting...
```

DOIT ESP32 DEVKIT V1, 80MHz, 921600, None on COM4

```
COM4  
|  
|  
| scan done  
| 2 networks found  
| 1: MEO-620B4B (-49)*  
| 2: MEO-WiFi (-50)  
|  
| scan start  
| scan done  
| 2 networks found  
| 1: MEO-620B4B (-48)*  
| 2: MEO-WiFi (-49)  
|  
|  
Autoscroll Both NL & CR 115200 baud Clear output
```

เสร็จแล้ว!

HOW TO...

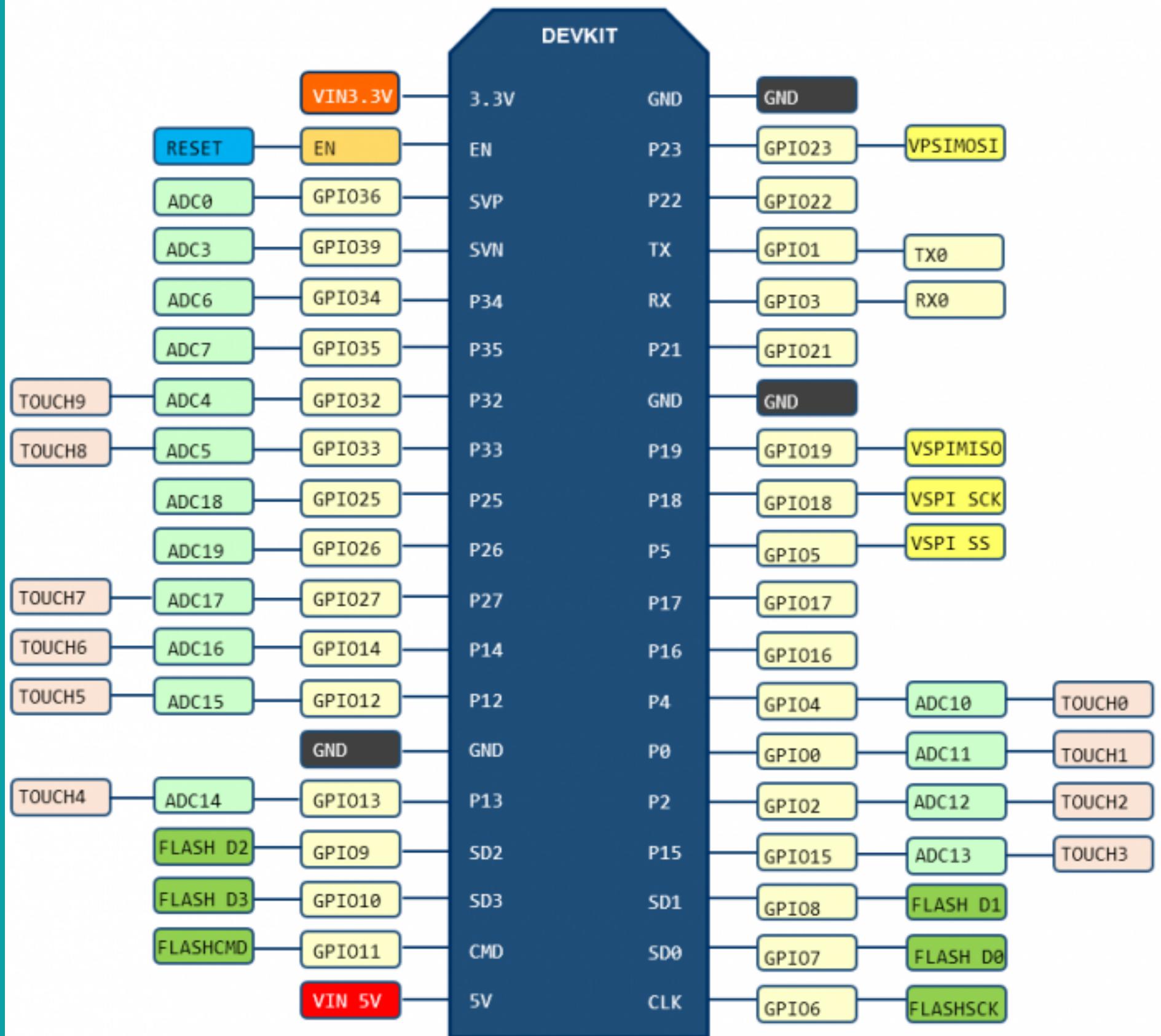


DIGITAL PORT

READ/WRITE

PIN DEFINITION

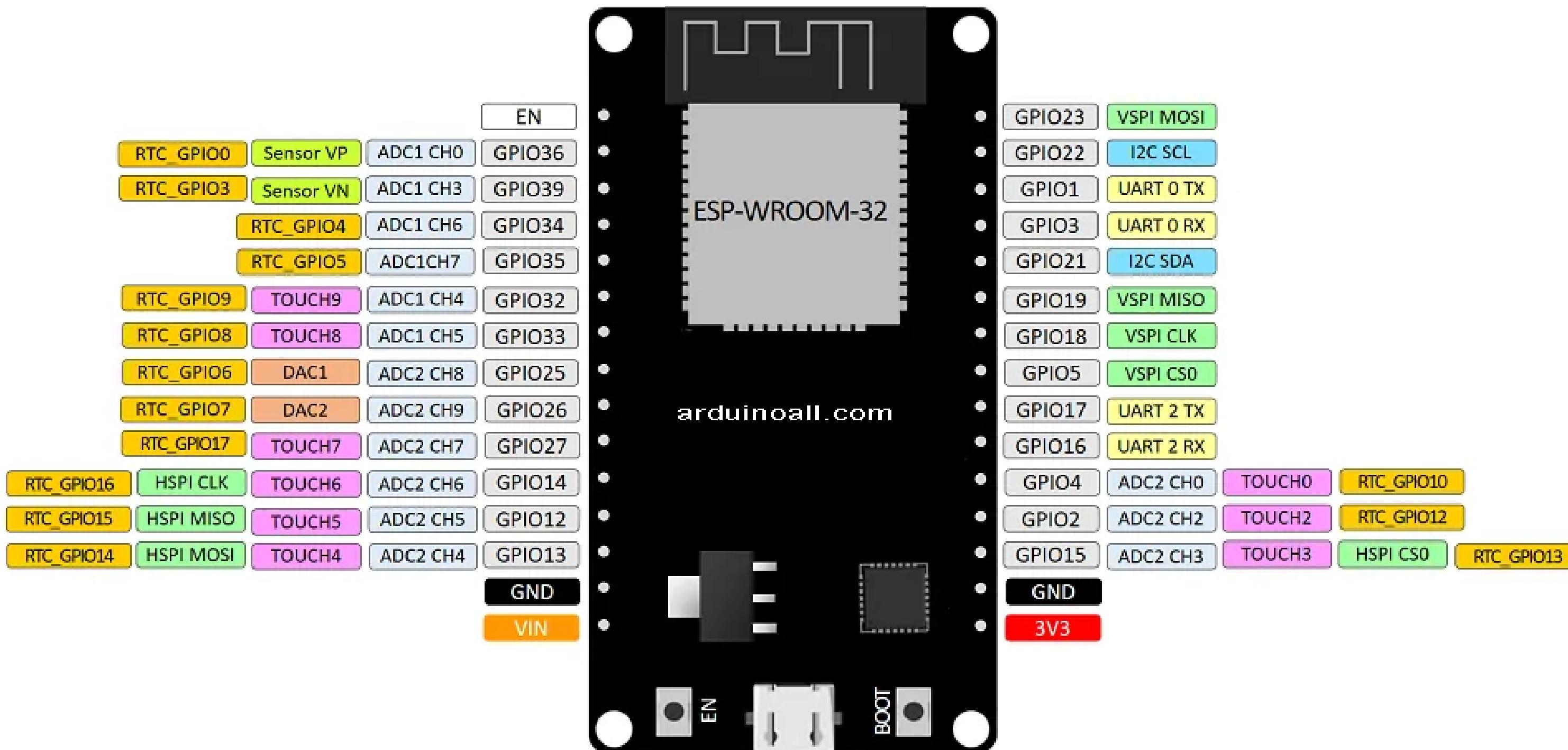
www.ai-thinker.com



NodeMCU-32S

ESP32 DEVKIT V1 - DOIT

version with 30 GPIOs



ESP32 Read Digital Inputs

First, set the GPIO you want to read as **INPUT**, using the **pinMode()** function as follows

```
pinMode(GPIO, INPUT);
```

To read a digital input, like a button, you use the **digitalRead()** function, that accepts as argument, the GPIO (int number) you are referring to.

```
digitalRead(GPIO);
```

All ESP32 GPIOs can be used as inputs, except GPIOs 6 to 11 (connected to the integrated SPI flash).

ESP32 Control Digital Outputs

First, you need set the GPIO you want to control as an **OUTPUT**. Use the **pinMode()** function as follows

```
pinMode(GPIO, OUTPUT);
```

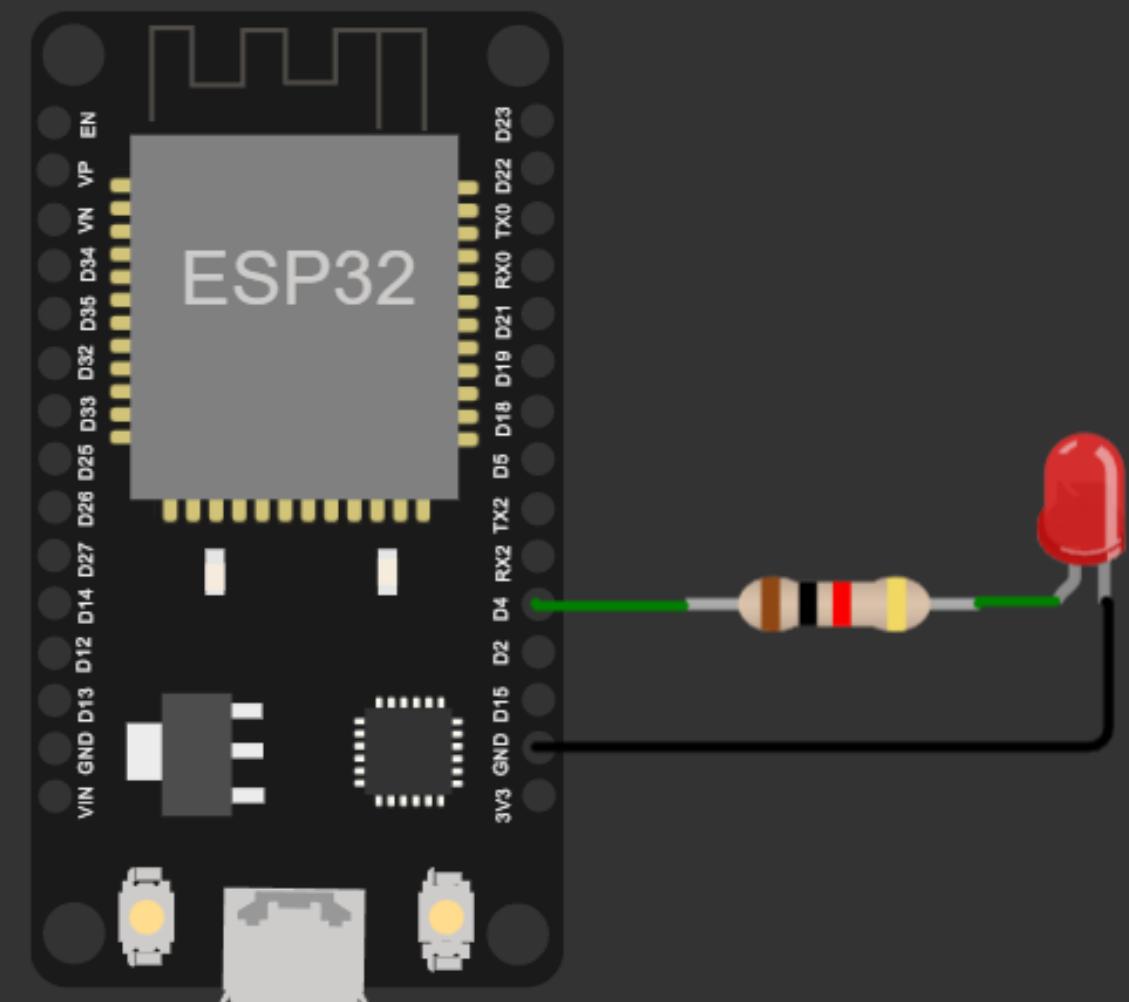
To control a digital output you just need to use the **digitalWrite()** function, that accepts as arguments, the GPIO (int number) you are referring to, and the state, either **HIGH** or **LOW**

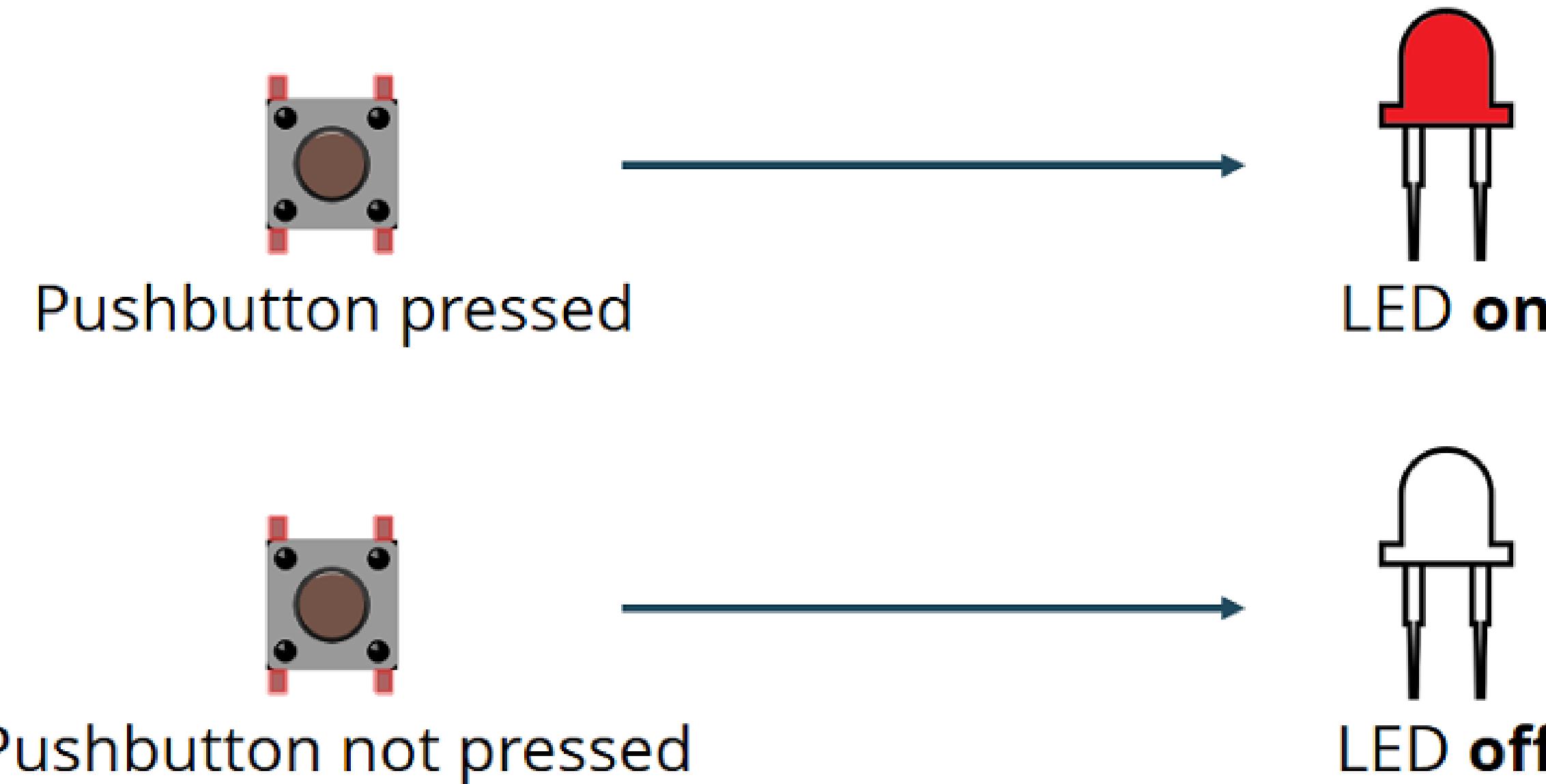
```
digitalWrite(GPIO, STATE);
```

- Coding

```
1 const int ledPin = 4; // GPIO 4
2
3 void setup() {
4     pinMode(ledPin, OUTPUT);
5 }
6
7 void loop() {
8     digitalWrite(ledPin, HIGH); // Turn on the LED
9     delay(1000); // Wait for 1 second
10    digitalWrite(ledPin, LOW); // Turn off the LED
11    delay(1000); // Wait for 1 second
12 }
```

- Schematic



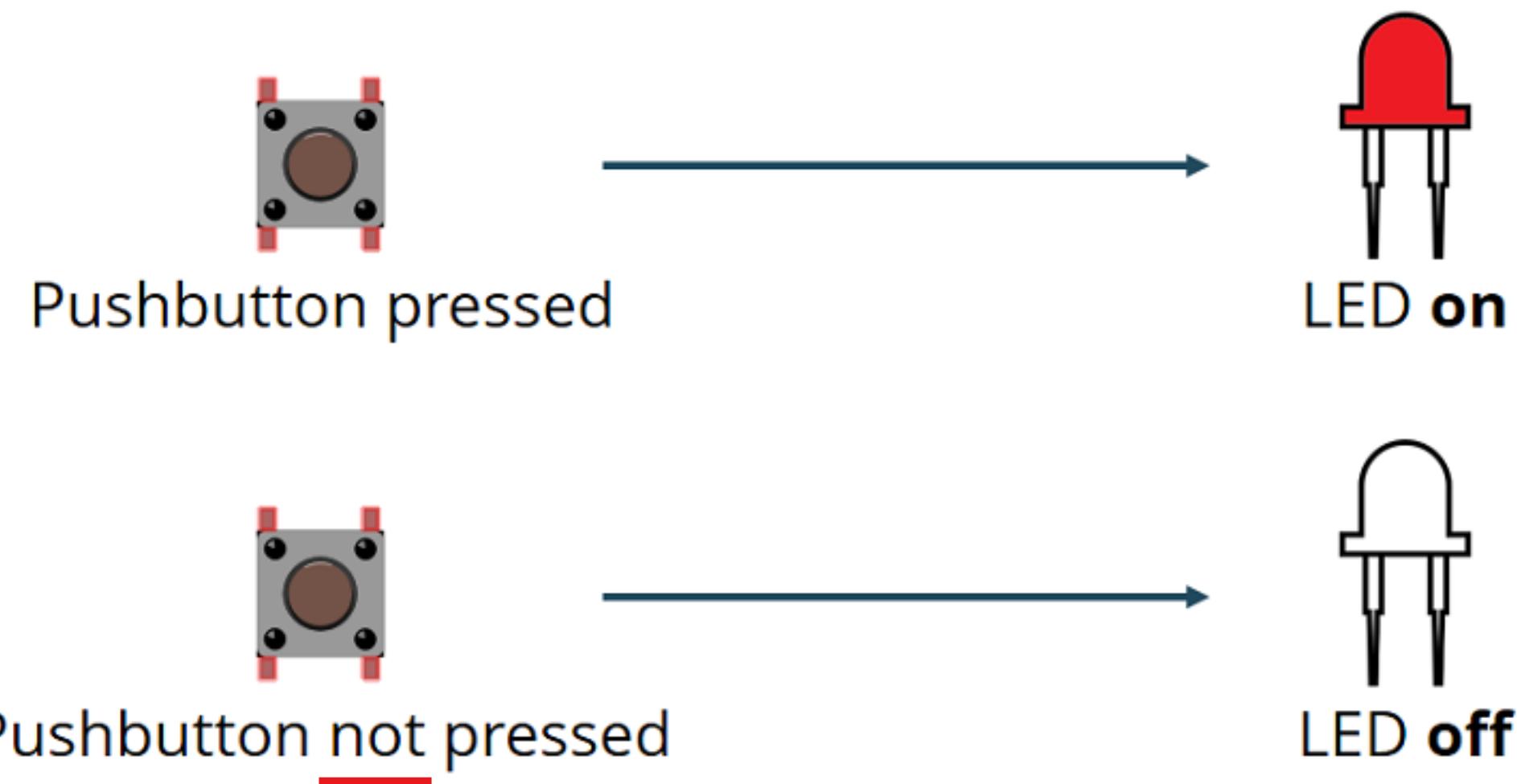


To read a digital input, like a button, you use the `digitalRead()` function, that accepts as argument, the GPIO (int number) you are referring to.

EXAMPLE

To show you how to use digital inputs and digital outputs, we'll build a simple project example with a pushbutton and an LED.

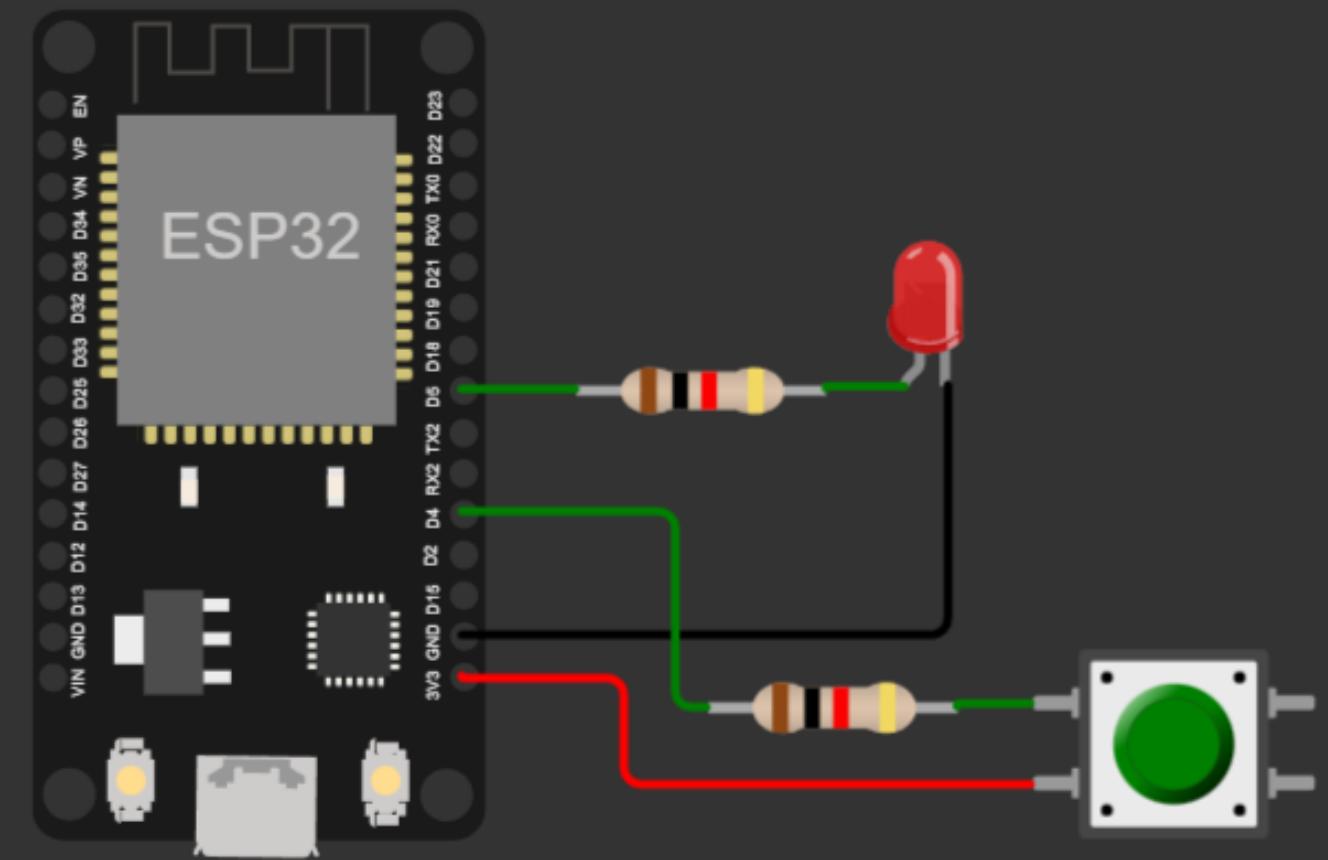
We'll read the state of the pushbutton and light up the LED.



- Coding

```
1 // set pin numbers
2 const int buttonPin = 4; // the number of the pushbutton pin
3 const int ledPin = 5; // the number of the LED pin
4
5 // variable for storing the pushbutton status
6 int buttonState = 0;
7
8 void setup() {
9     Serial.begin(115200);
10    // initialize the pushbutton pin as an input
11    pinMode(buttonPin, INPUT);
12    // initialize the LED pin as an output
13    pinMode(ledPin, OUTPUT);
14 }
15
16 void loop() {
17    // read the state of the pushbutton value
18    buttonState = digitalRead(buttonPin);
19    Serial.println(buttonState);
20    // check if the pushbutton is pressed.
21    // if it is, the buttonState is HIGH
22    if (buttonState == HIGH) {
23        // turn LED on
24        digitalWrite(ledPin, HIGH);
25    } else {
26        // turn LED off
27        digitalWrite(ledPin, LOW);
28    }
29 }
```

- Schematic



Before proceeding, you need to assemble a circuit with an LED and a pushbutton. We'll connect the LED to GPIO 5 and the pushbutton to GPIO 4.

ANALOG PORT

READ/WRITE

ESP32 ADC – Read Analog Values

- **Analog Inputs (ADC)**

Reading an analog value with the ESP32 means you can measure varying voltage levels between 0 V and 3.3 V. The voltage measured is then assigned to a value between 0 and 4095, in which 0 V corresponds to 0, and 3.3 V corresponds to 4095. Any voltage between 0 V and 3.3 V will be given the corresponding value in between.

```
const int potPin = 34;
```

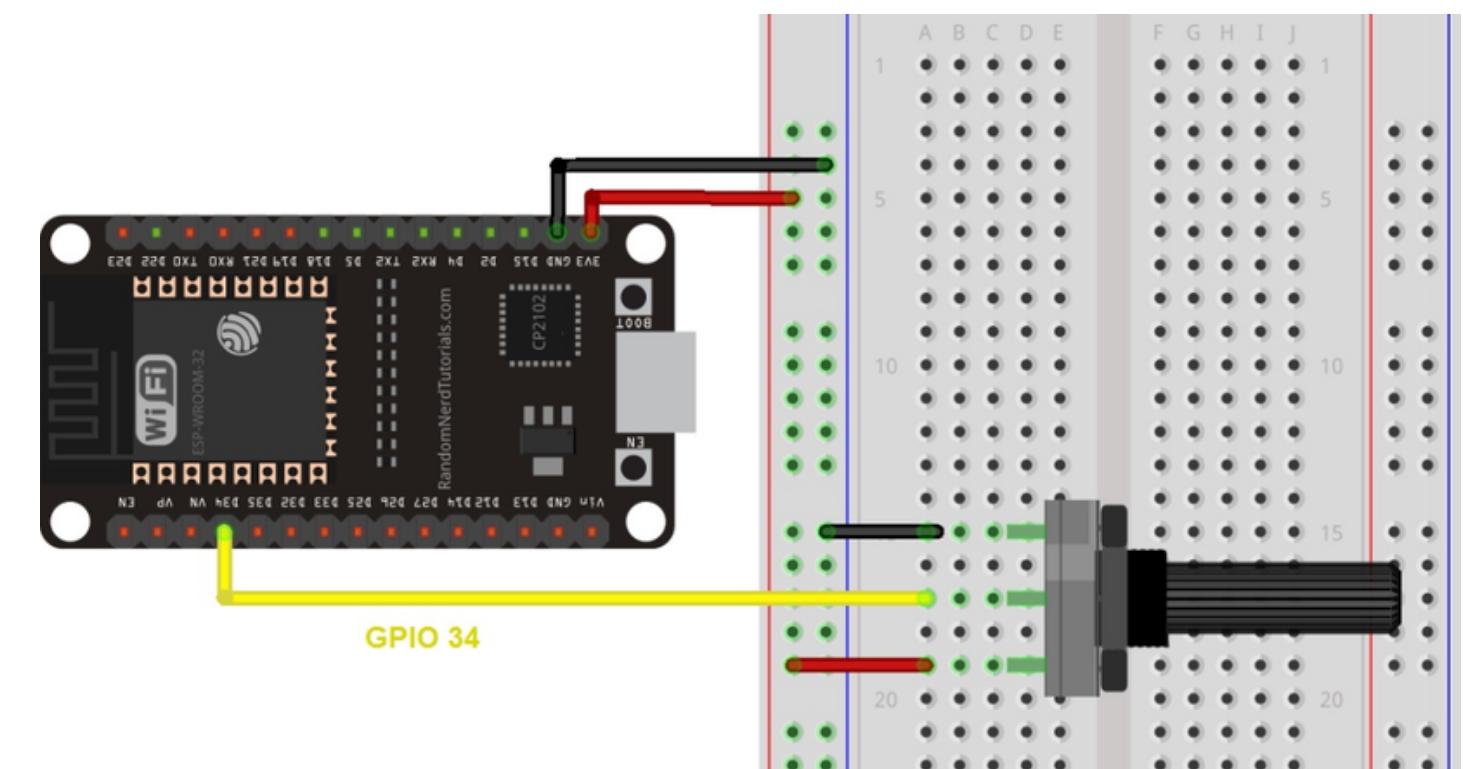
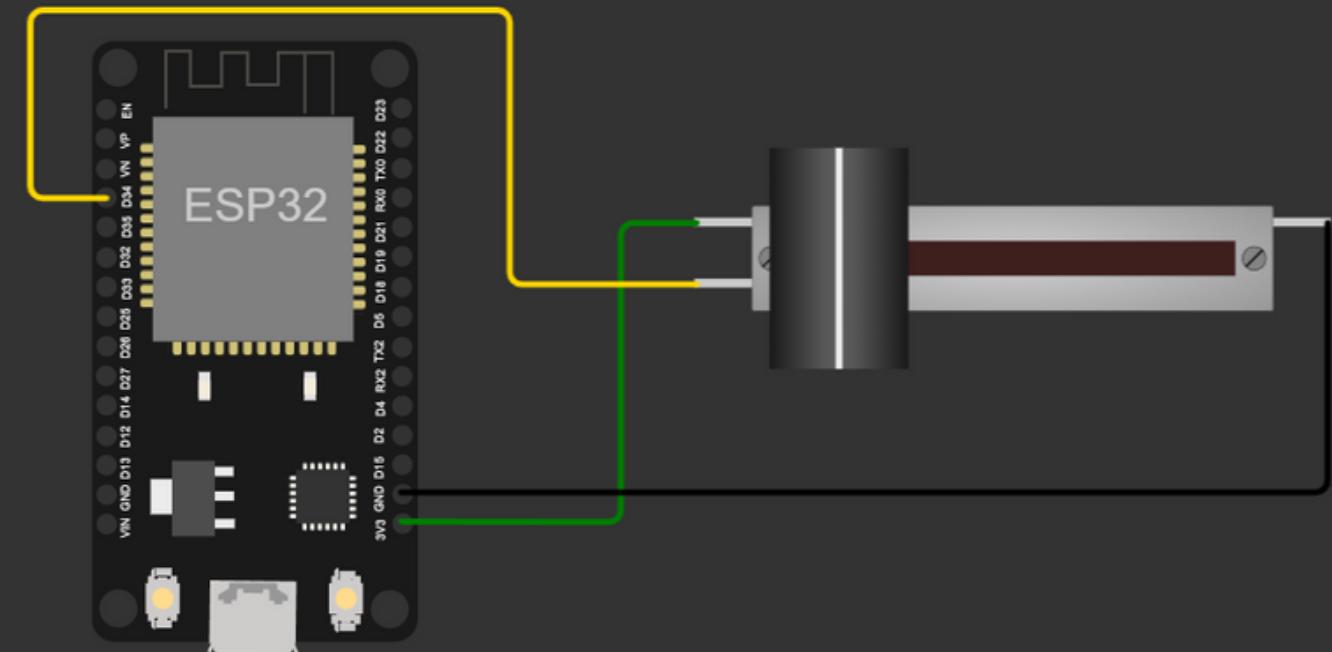
In the **setup()**, initialize a serial communication at a baud rate of 115200.

```
Serial.begin(115200);
```

- Coding

```
1 const int potPin = 34;  
2  
3 // variable for storing the potentiometer value  
4 int potValue = 0;  
5  
6 void setup() {  
7     Serial.begin(115200);  
8     delay(1000);  
9 }  
10  
11 void loop() {  
12     // Reading potentiometer value  
13     potValue = analogRead(potPin);  
14     Serial.println(potValue);  
15     delay(500);  
16 }
```

- Schematic



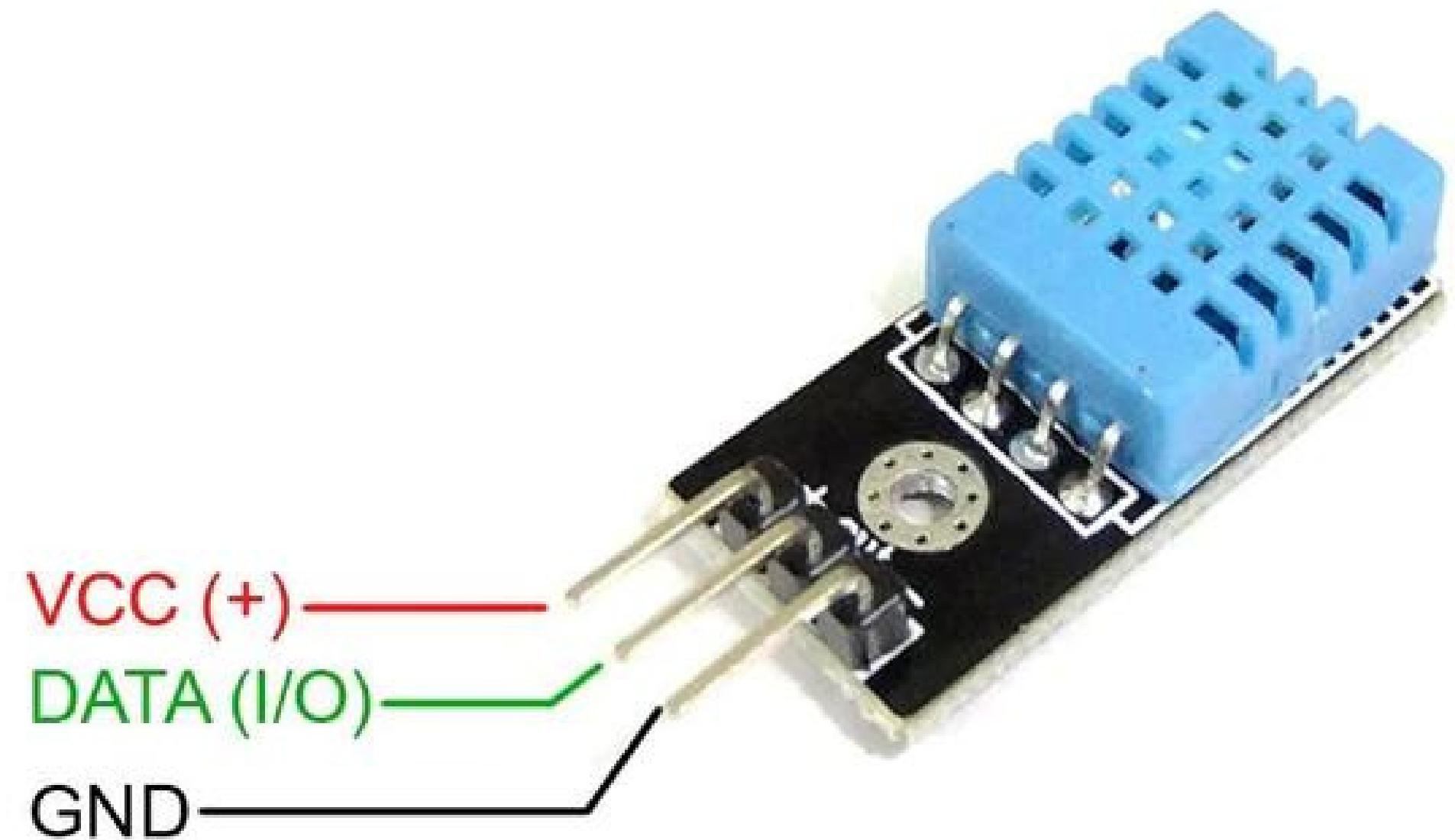
SENSOR

LED / RESISTOR / SWITCH / TEMP S. / INFRARED S. / SERVO MOTOR

DHT11 - Temperature and Humidity Sensor

The DHT11 sensors are used to measure temperature and relative humidity. These sensors contain a chip that does analog to digital conversion and spit out a digital signal with the temperature and humidity..

DHT11	
Operating Voltage	3 to 5V
Max Operating Current	2.5mA max
Humidity Range	20-80% / 5%
Temperature Range	0-50°C / ± 2°C
Sampling Rate	1 Hz (reading every second)
Body size	15.5mm x 12mm x 5.5mm
Advantage	Ultra low cost



DHTtester | Arduino 1.8.19

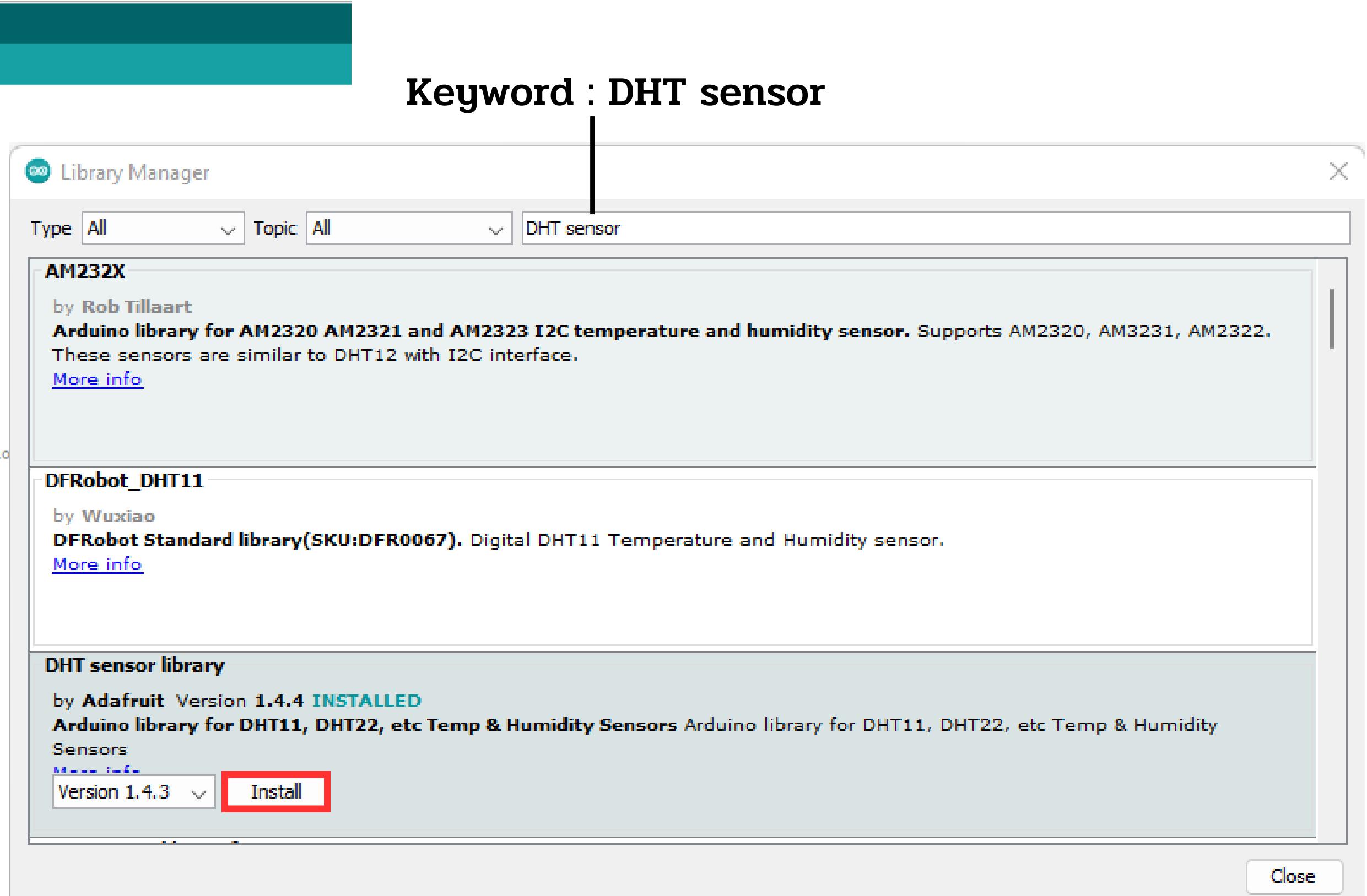
File Edit Sketch Tools Help

DHTtester §

```
#include "DHT.h"
#define DHTPI
#define DHTTY
DHT dht(DHTPI);

void setup() {
  Serial.begin()
  Serial.print
  dht.begin()
}

void loop() {
  // Wait a f
  delay(2000)
  // Reading
  // Sensor r
  float h = d
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
}
```



ତିଦତ୍ତଙ୍କ

DHT11-Library

Close

• Coding

```
#include "DHT.h"
#define DHTPIN 4      // Digital pin connected to the DHT sensor
#define DHTTYPE DHT11 // DHT 11
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));
  dht.begin();
}


```

```
void loop() {
  // Wait a few seconds between measurements.
  delay(2000);

  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }

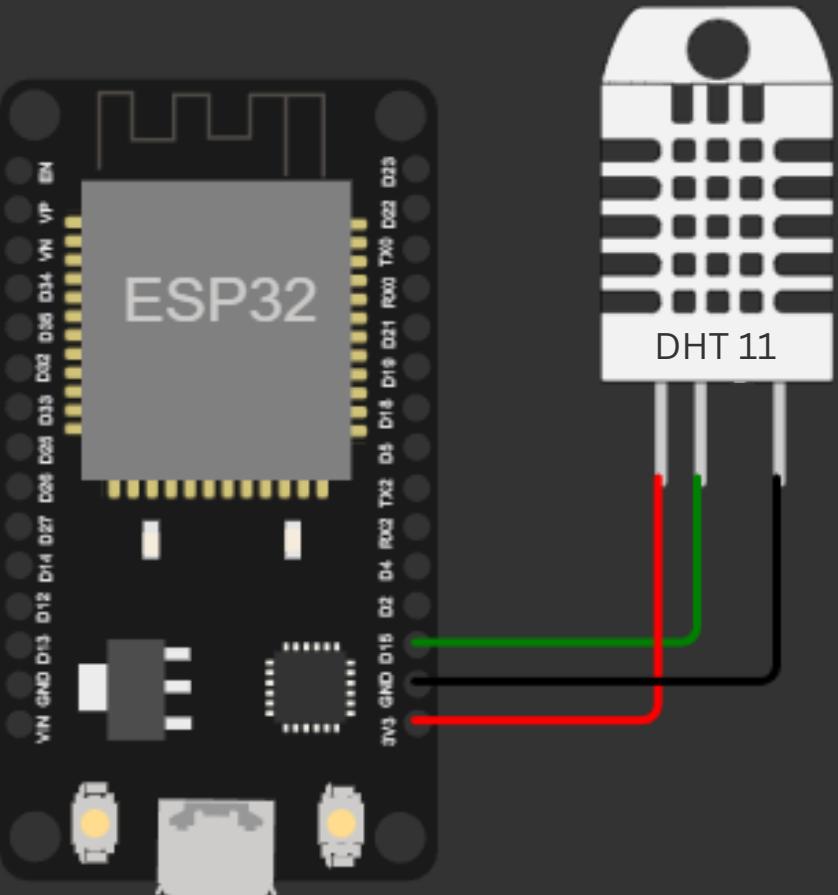
}
```

```
// Compute heat index in Fahrenheit (the default)
float hif = dht.computeHeatIndex(f, h);
// Compute heat index in Celsius (isFahrenheit = false)
float hic = dht.computeHeatIndex(t, h, false);

Serial.print(F("Humidity: "));
Serial.print(h);
Serial.print(F("% Temperature: "));
Serial.print(t);
Serial.print(F("°C "));
Serial.print(f);
Serial.print(F("°F Heat index: "));
Serial.print(hic);
Serial.print(F("°C "));
Serial.print(hif);
Serial.println(F("°F"));

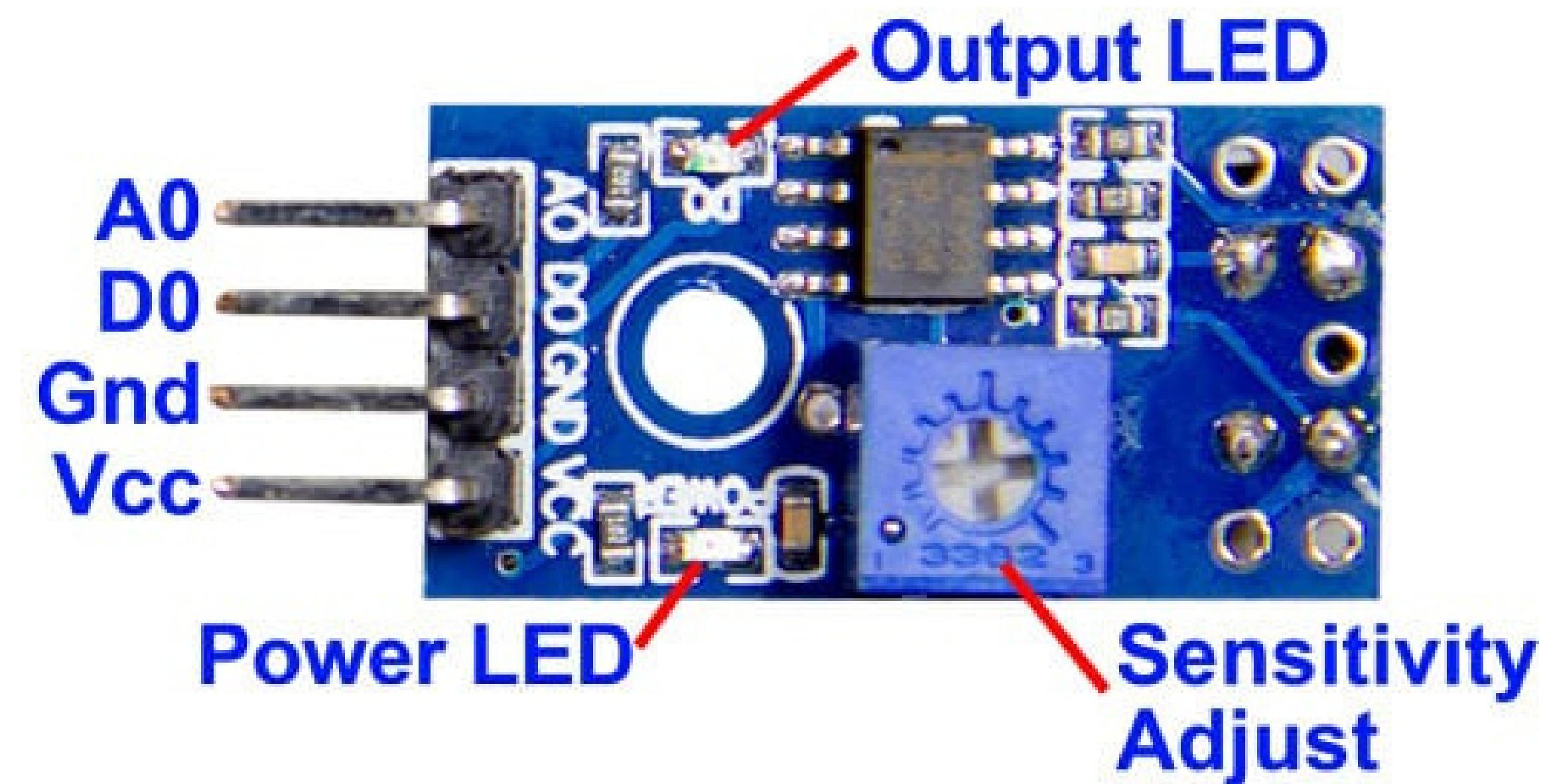
}
```

• Schematic



IR - Infrared Sensor

An **infrared proximity sensor** or IR Sensor is an electronic device that emits infrared lights to sense some aspect of the surroundings and can be employed to detect the motion of an object. As this is a passive sensor, it can only measure infrared radiation.



• Coding

```

void setup() {
    pinMode(D1, INPUT);
    Serial.begin(9600);
}

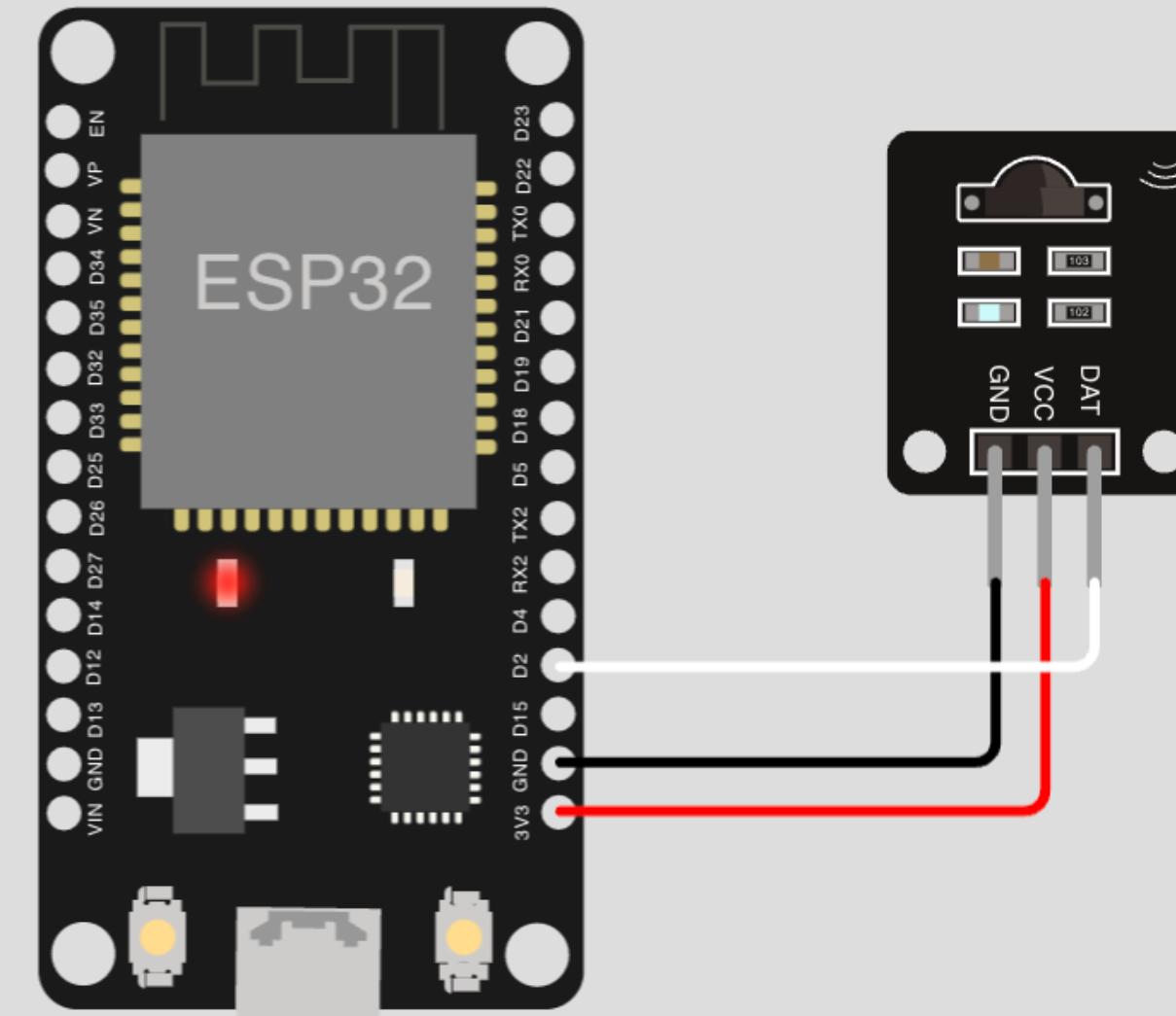
void loop() {
    if(digitalRead(D1)==LOW) {
        Serial.println("Deteact");
        delay(1000);
    } else {
        Serial.println("Unknown");
        delay(1000);
    }
}

```

• Output & Schematic

COM5

20:42:00.746 -> Unknown // เมื่อไม่ถูกสัมผัส
 20:42:01.727 -> Deteact // เมื่อถูกสัมผัส





A woman is sleeping peacefully in a bed with white linens. A glowing orange nightstand lamp is visible on the right side of the frame. The background is dark, suggesting it's nighttime.

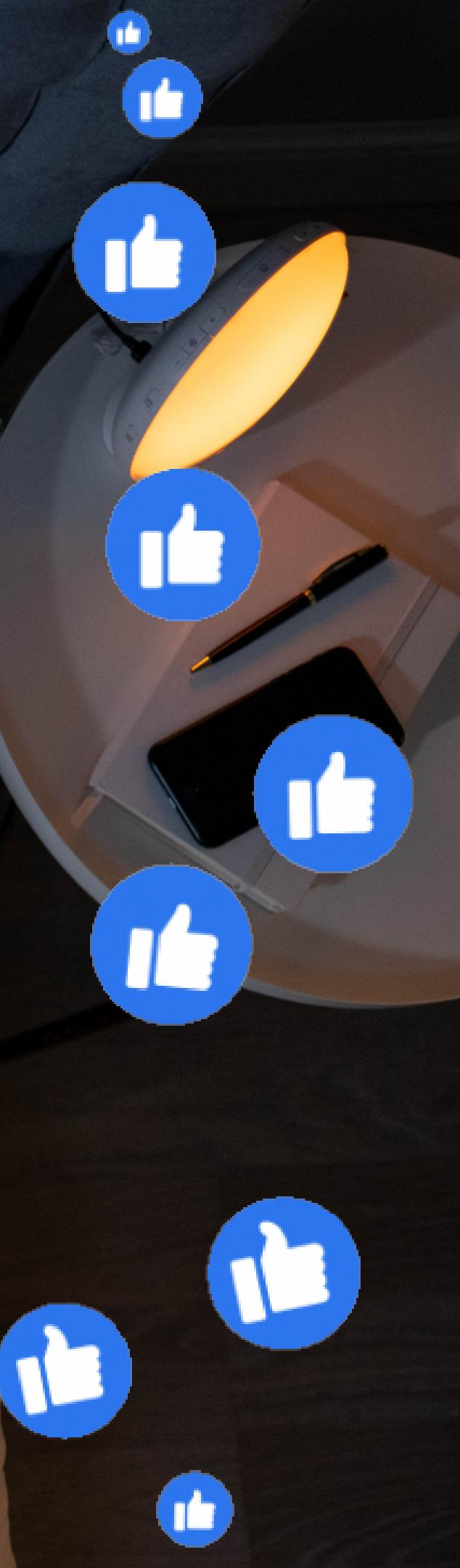
THANK YOU FOR THIS DAY



Follow

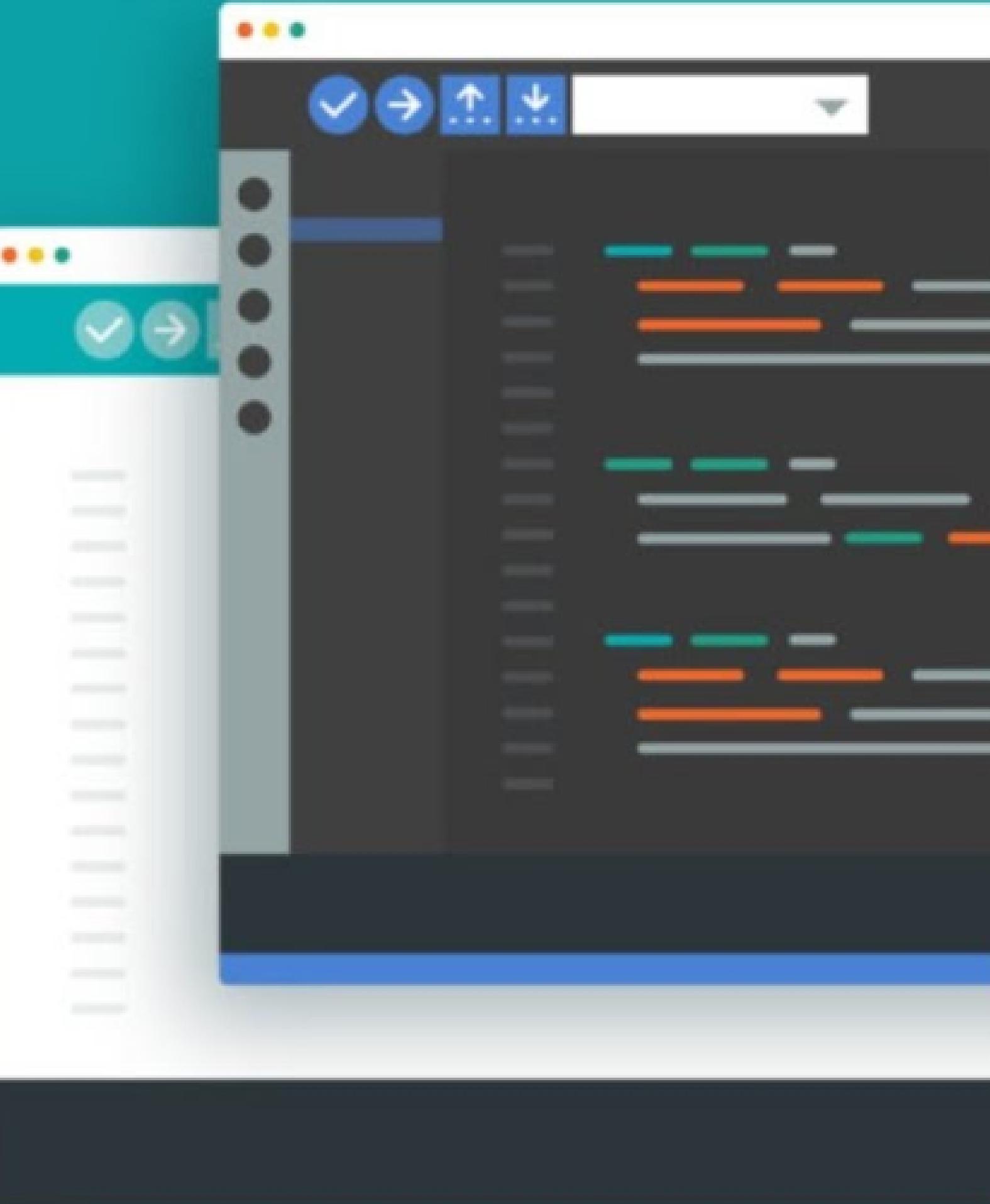


Share



INTERNET OF THINGS

DAY 2



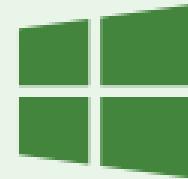
NODE-RED

เครื่องมือสำหรับนักพัฒนาโปรแกรมในการเชื่อมต่ออุปกรณ์hardwareเข้ากับ API

Setting Tools

LTS

Recommended For Most Users



Windows Installer

node-v18.16.0-x64.msi



macOS Installer

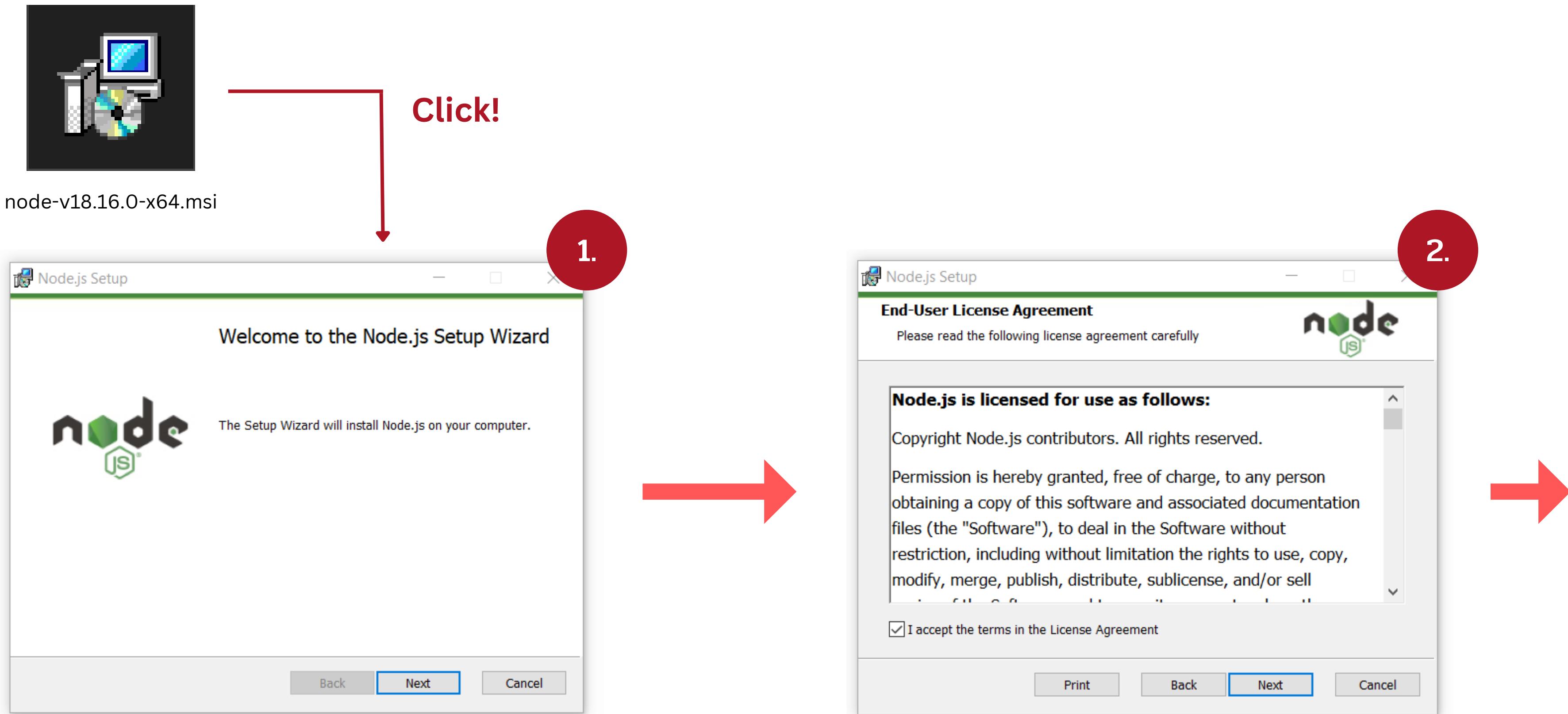
node-v18.16.0.pkg

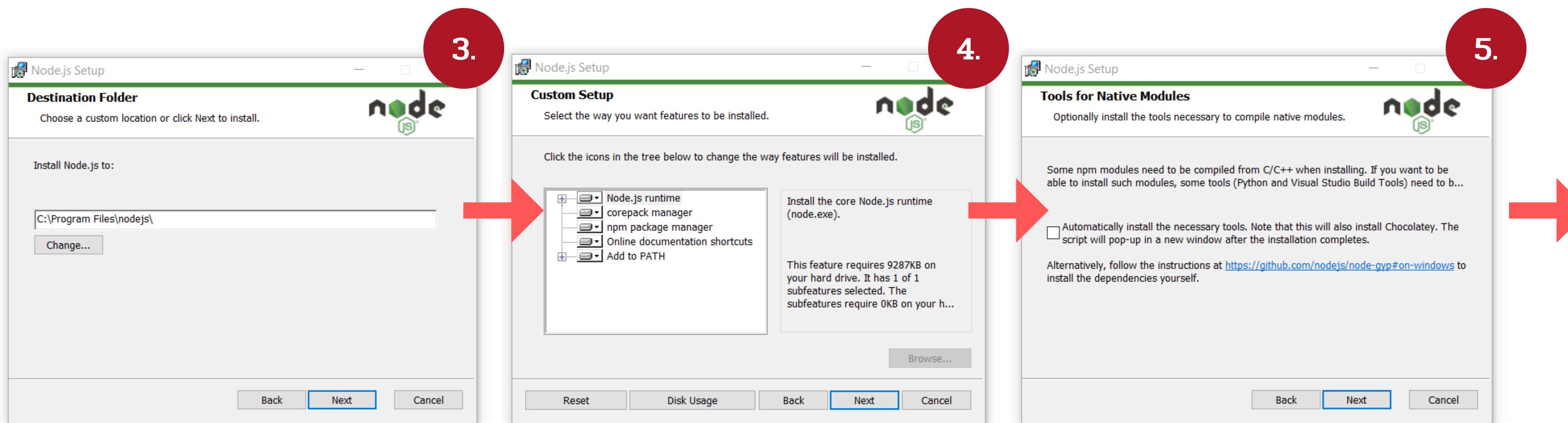


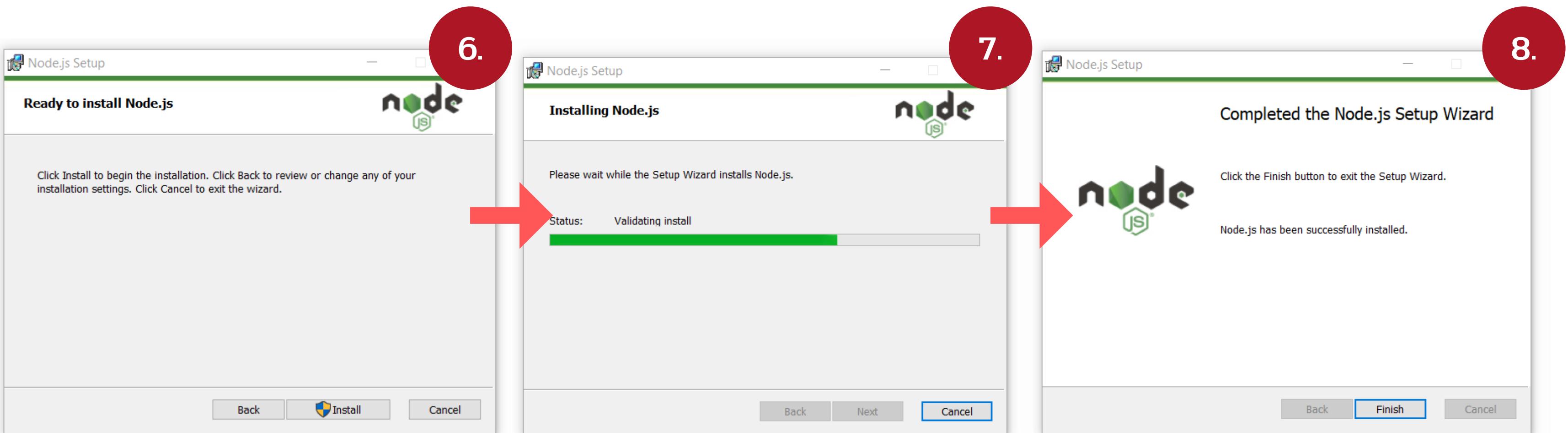
Source Code

node-v18.16.0.tar.gz

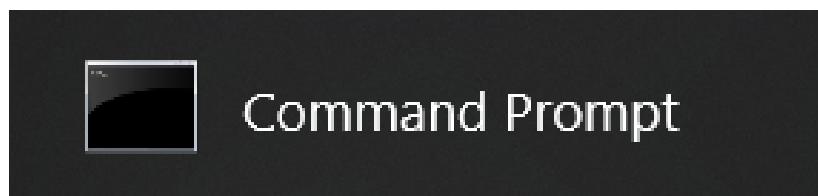
กดดาวน์โหลด







เปิด Command Prompt (กดคันหาที่แถบเมนูด้านล่าง)



เปิด cmd

```
Microsoft Windows [Version 10.0.19045.2965]
(c) Microsoft Corporation. All rights reserved.

C:\Users\khamp>
```

node -v

```
C:\Users\khamp>node -v
v18.16.0
```

npm -v

```
C:\Users\khamp>npm -v
9.5.1
```

npm install node-red -g --unsafe-perm

```
C:\Users\khamp>npm install node-red -g --unsafe-perm
[██████████] | reify:leven: timing reifyNode:node_modules/node-red/node_modules/end-of
```

รอโหลดจ้าาา



```
added 293 packages in 23s
```

```
41 packages are looking for funding
  run `npm fund` for details
```

```
npm notice
```

```
npm notice New minor version of npm available! 9.5.1 => 9.6.7
npm notice Changelog: https://github.com/npm/cli/releases/tag/v9.6.7
npm notice Run npm install -g npm@9.6.7 to update!
npm notice
```

```
C:\Users\khamp>node-red
26 May 22:45:56 - [info]

Welcome to Node-RED
=====

26 May 22:45:56 - [info] Node-RED version: v3.0.2
26 May 22:45:56 - [info] Node.js version: v18.16.0
26 May 22:45:56 - [info] Windows_NT 10.0.19045 x64 LE
26 May 22:45:58 - [info] Loading palette nodes
26 May 22:45:59 - [info] Settings file : C:\Users\khamp\.node-red\settings.js
26 May 22:45:59 - [info] Context store : 'default' [module=memory]
26 May 22:45:59 - [info] User directory : C:\Users\khamp\.node-red
26 May 22:45:59 - [warn] Projects disabled : editorTheme.projects.enabled=false
26 May 22:45:59 - [info] Flows file : C:\Users\khamp\.node-red\flows.json
26 May 22:45:59 - [info] Creating new flow file
26 May 22:45:59 - [warn]

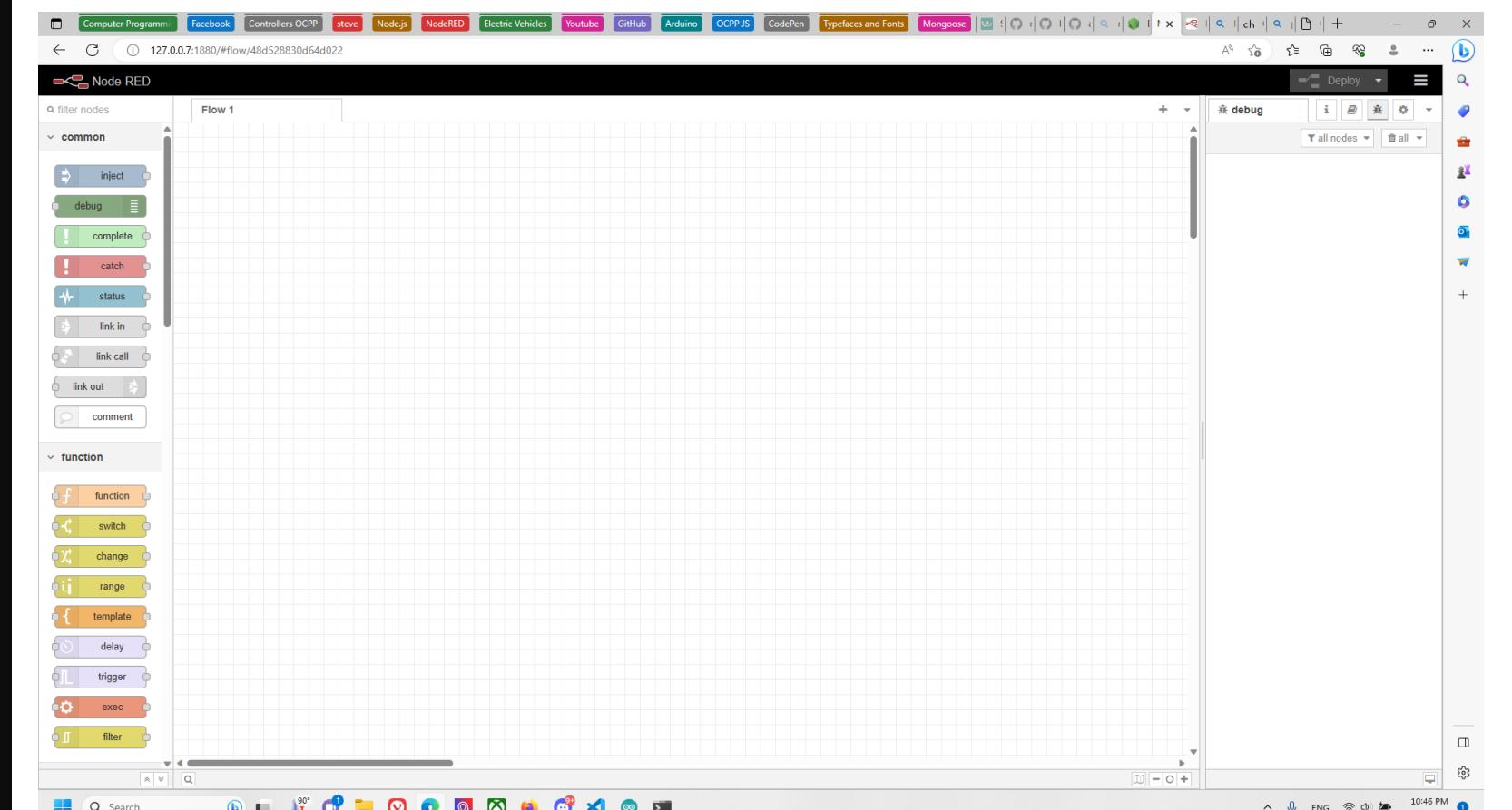
-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

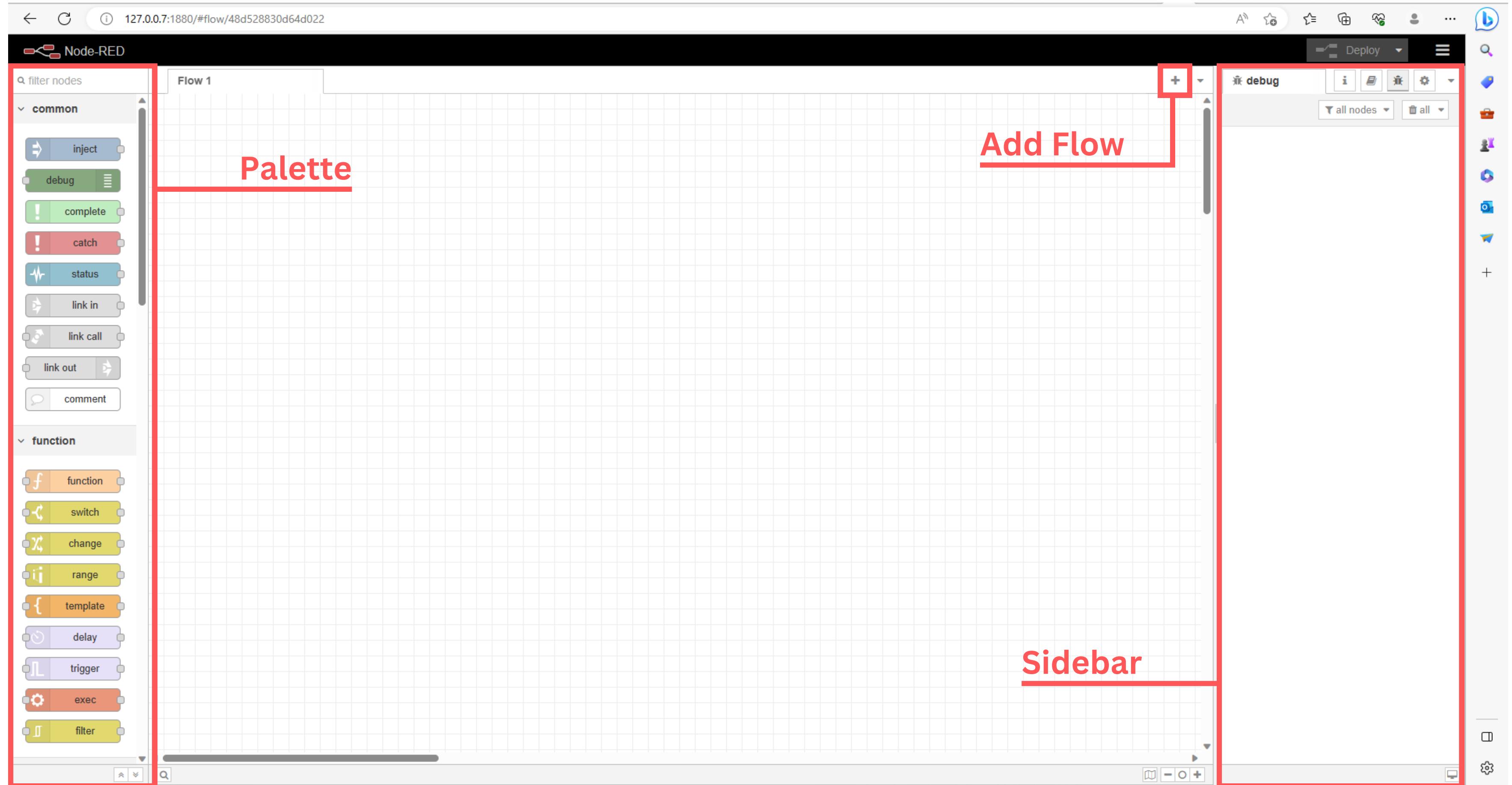
You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.

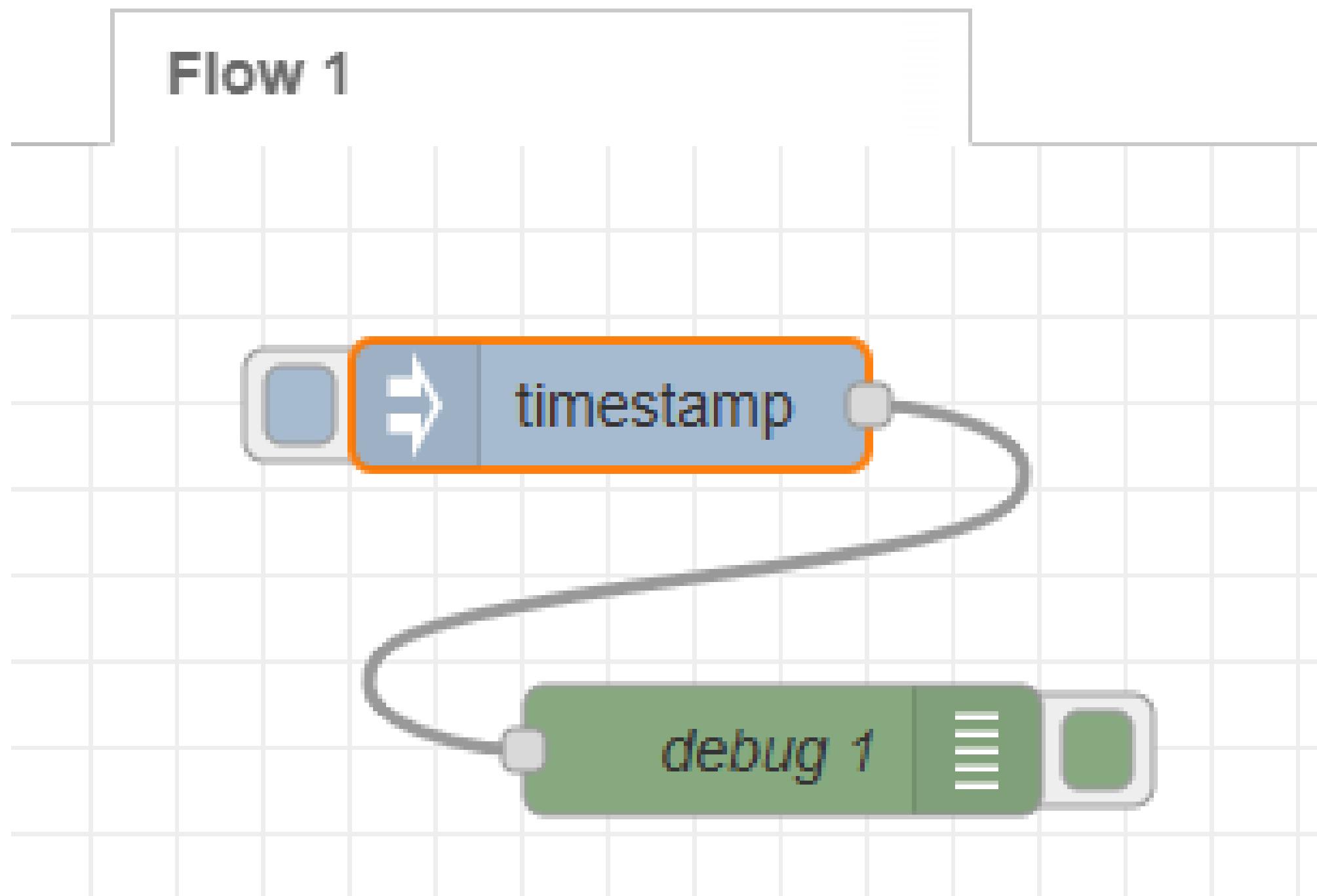
-----
26 May 22:45:59 - [info] Server now running at http://127.0.0.1:1880/
26 May 22:45:59 - [warn] Encrypted credentials not found
26 May 22:45:59 - [info] Starting flows
26 May 22:45:59 - [info] Started flows
```

node-red

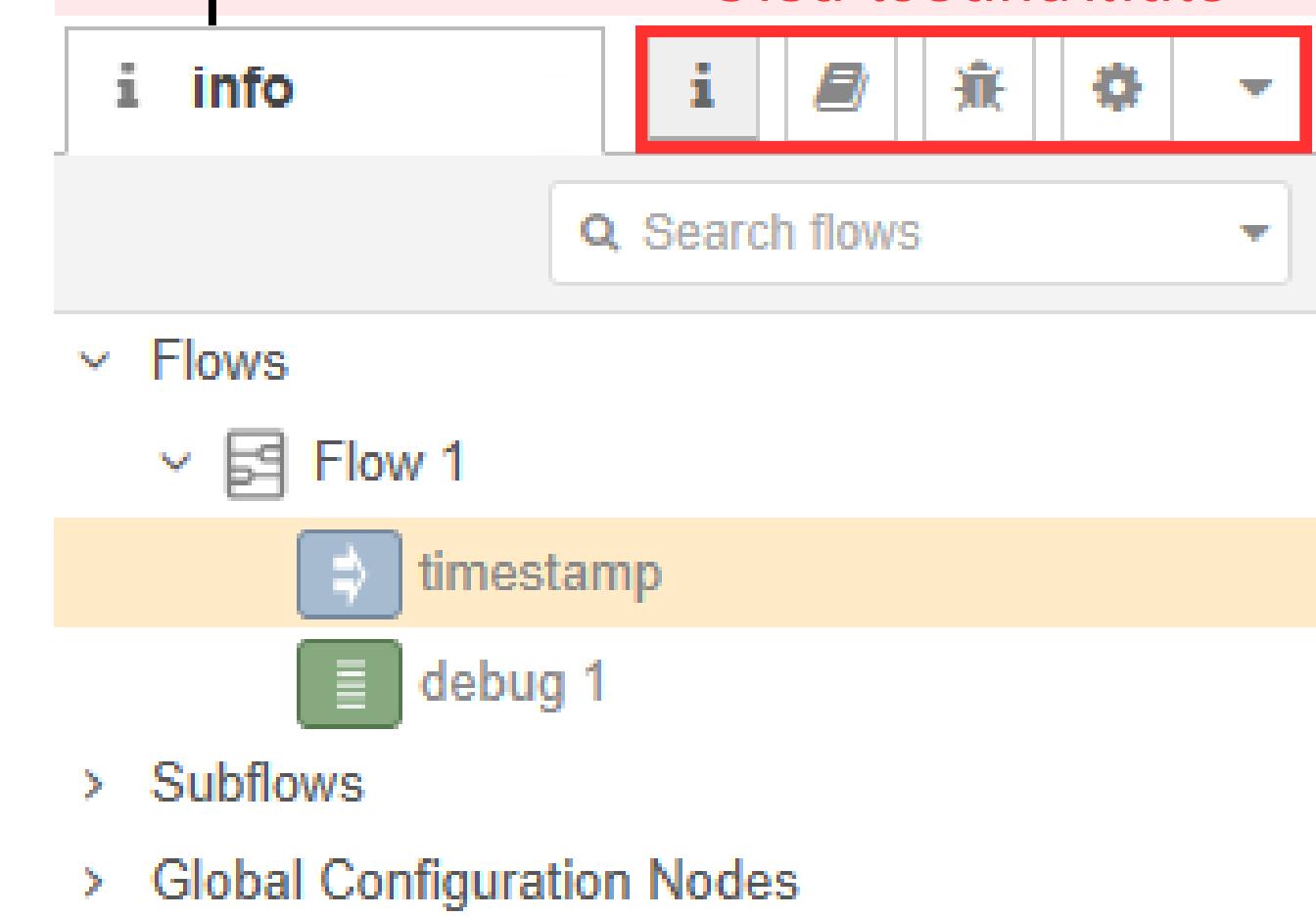


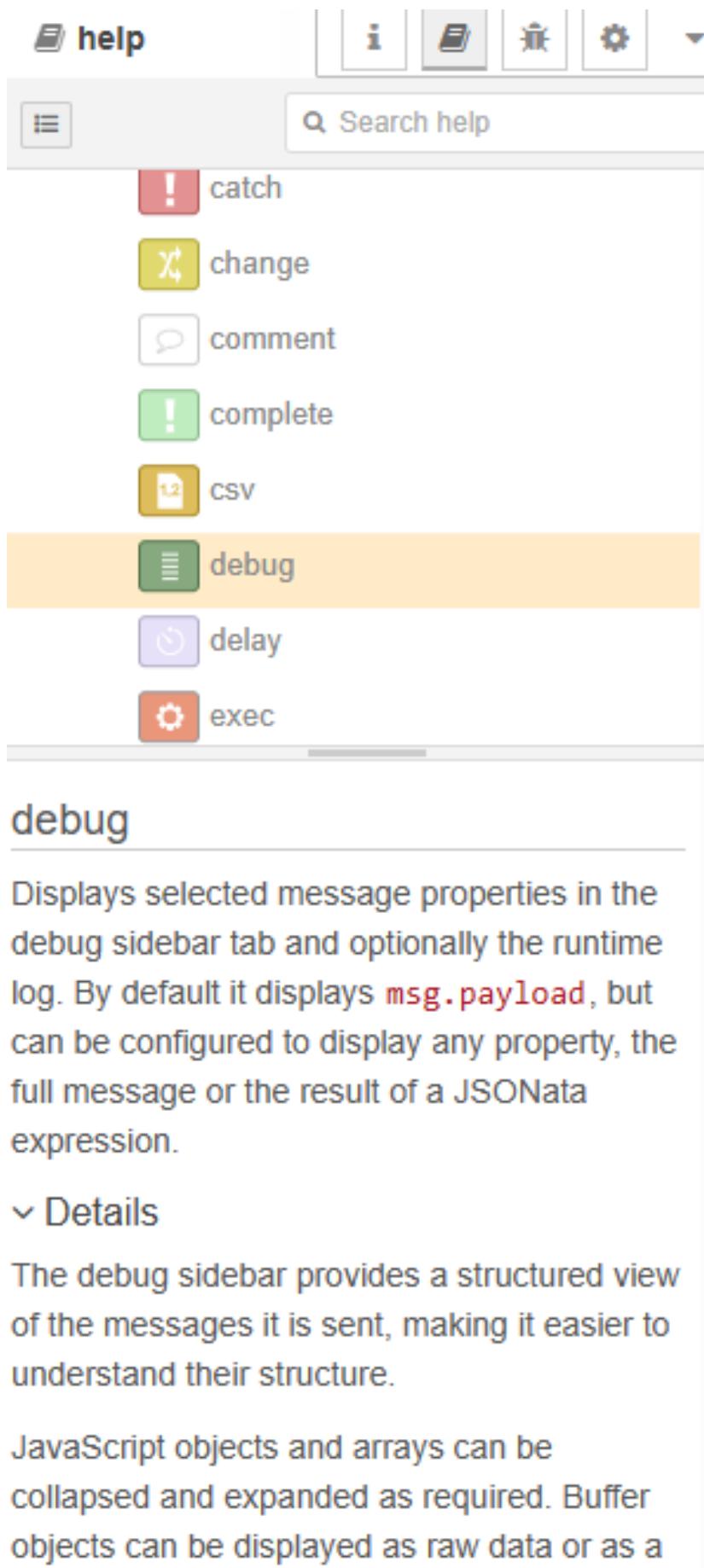
พิมพ์ **127.0.0.1:1880/**
ที่ Browser (เช่น Google Chrome)





info : รายละเอียดว่า "เราสร้างอะไรไปบ้าง"





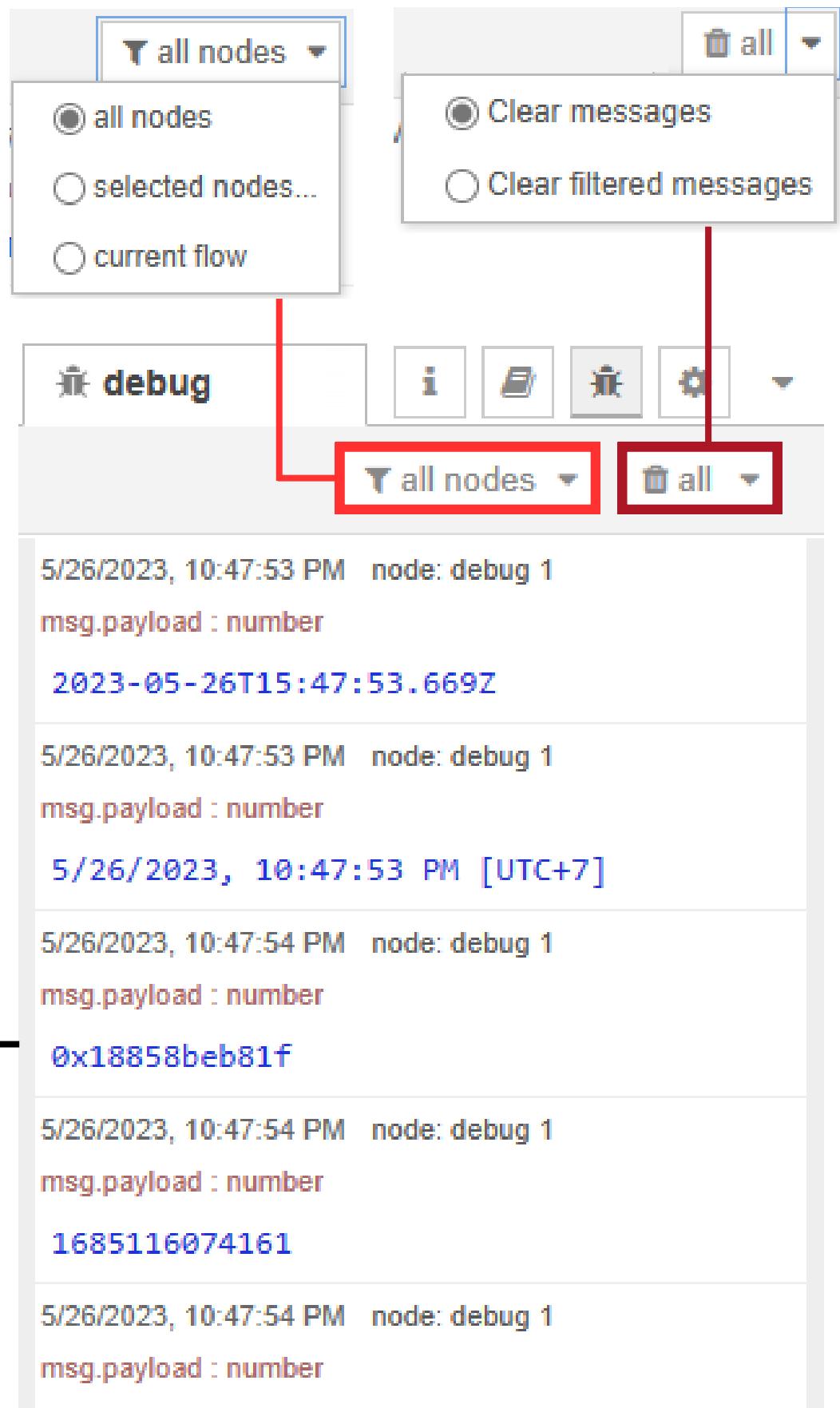
info : รายละเอียดว่า "เราสร้างอะไรไปบ้าง"

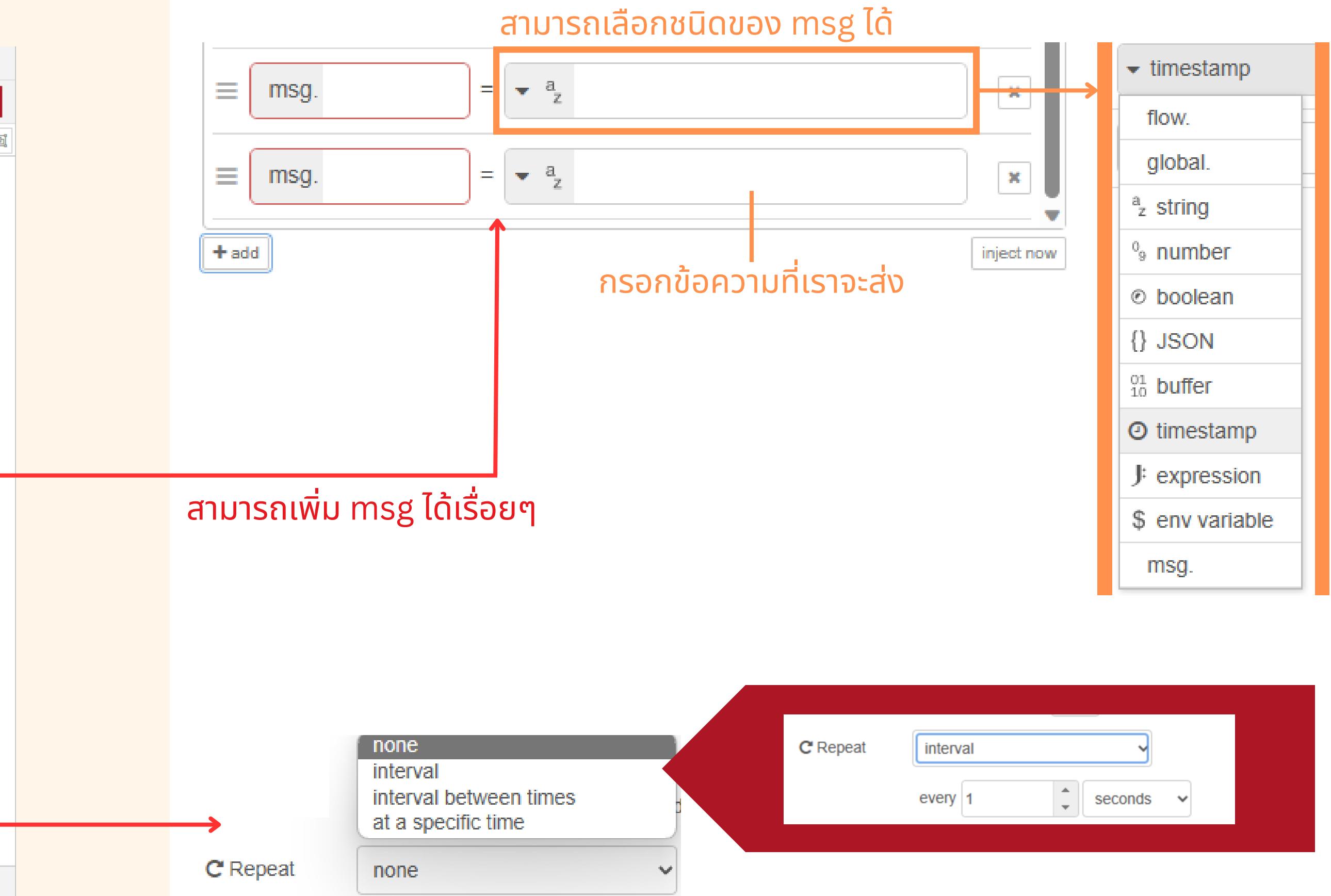
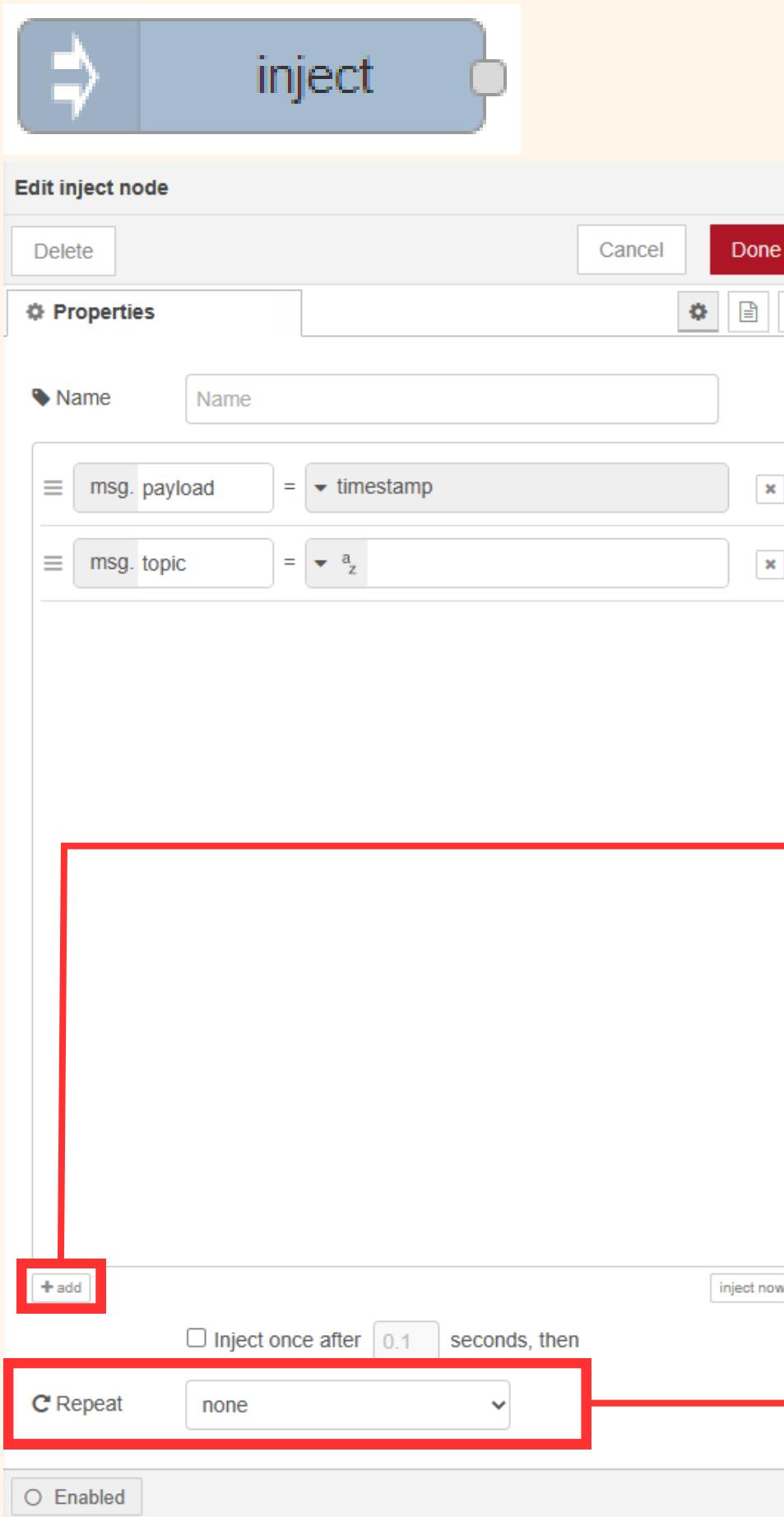
config :
ดูการตั้งค่าของ node

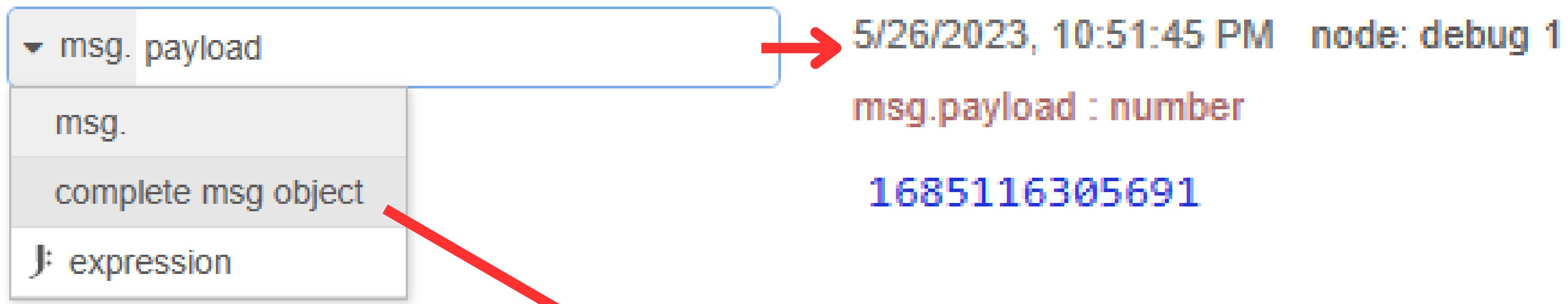


help : คู่มือประกอบวิธีใช้

debug :
หน้าต่างแสดงผลที่ส่งค่า







5/26/2023, 10:51:58 PM node: debug 1

msg : Object

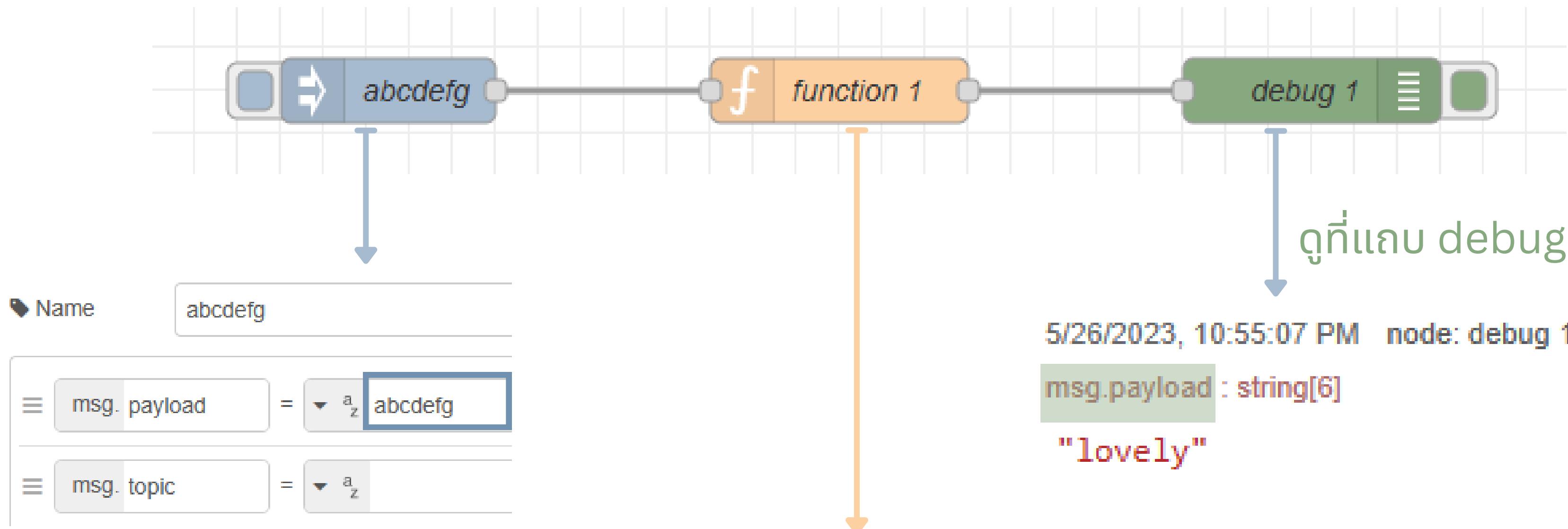
▼ object

_msgid: "80d569b16d87d994"

payload: 1685116318622

topic: ""

Examples

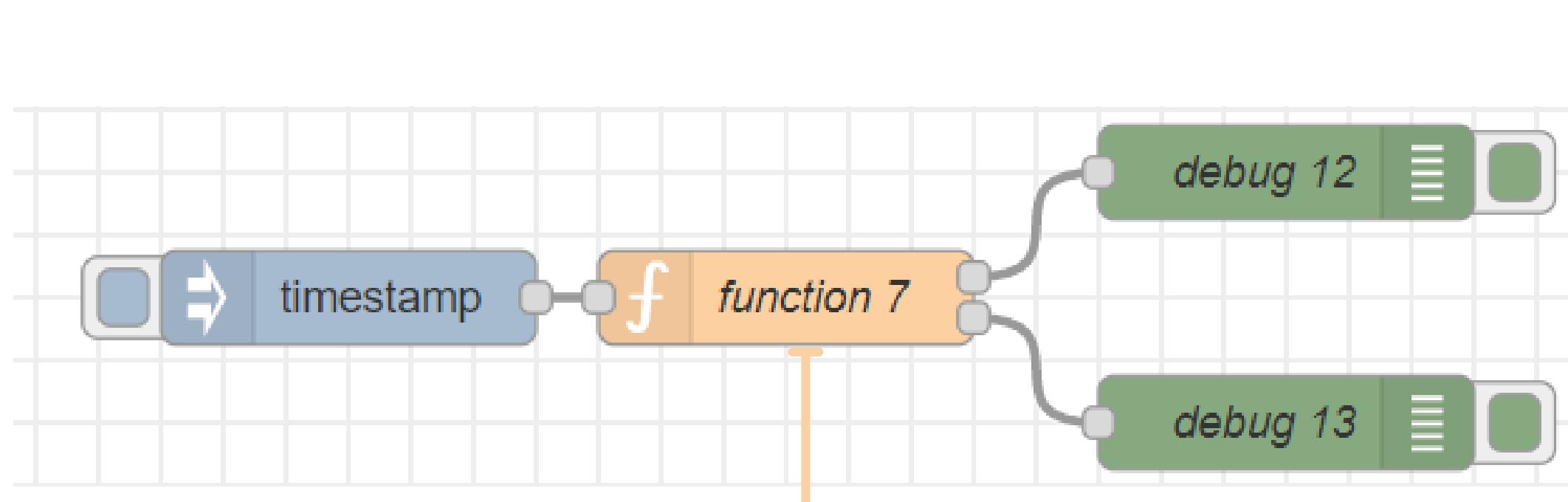


ระบุ key ที่เราจะเปลี่ยนข้อความ

Name: function 1

Setup On Start On Message **On Message** On Stop

```
1 msg.payload = "lovely";  
2 return msg;
```



ระบุ key แล้วใส่ข้อความ

Name: function 7

On Message

```

1 var msg1 = { payload: "first out of output 1" };
2 var msg2 = { payload: "second out of output 1" };
3 var msg3 = { payload: "third out of output 1" };
4 var msg4 = { payload: "only message from output 2" };
5 return [[msg1, msg2, msg3], msg4];
  
```

Outputs: 2

Setup

On Start

On Message

On Stop

debug 12 **debug 13**

debug

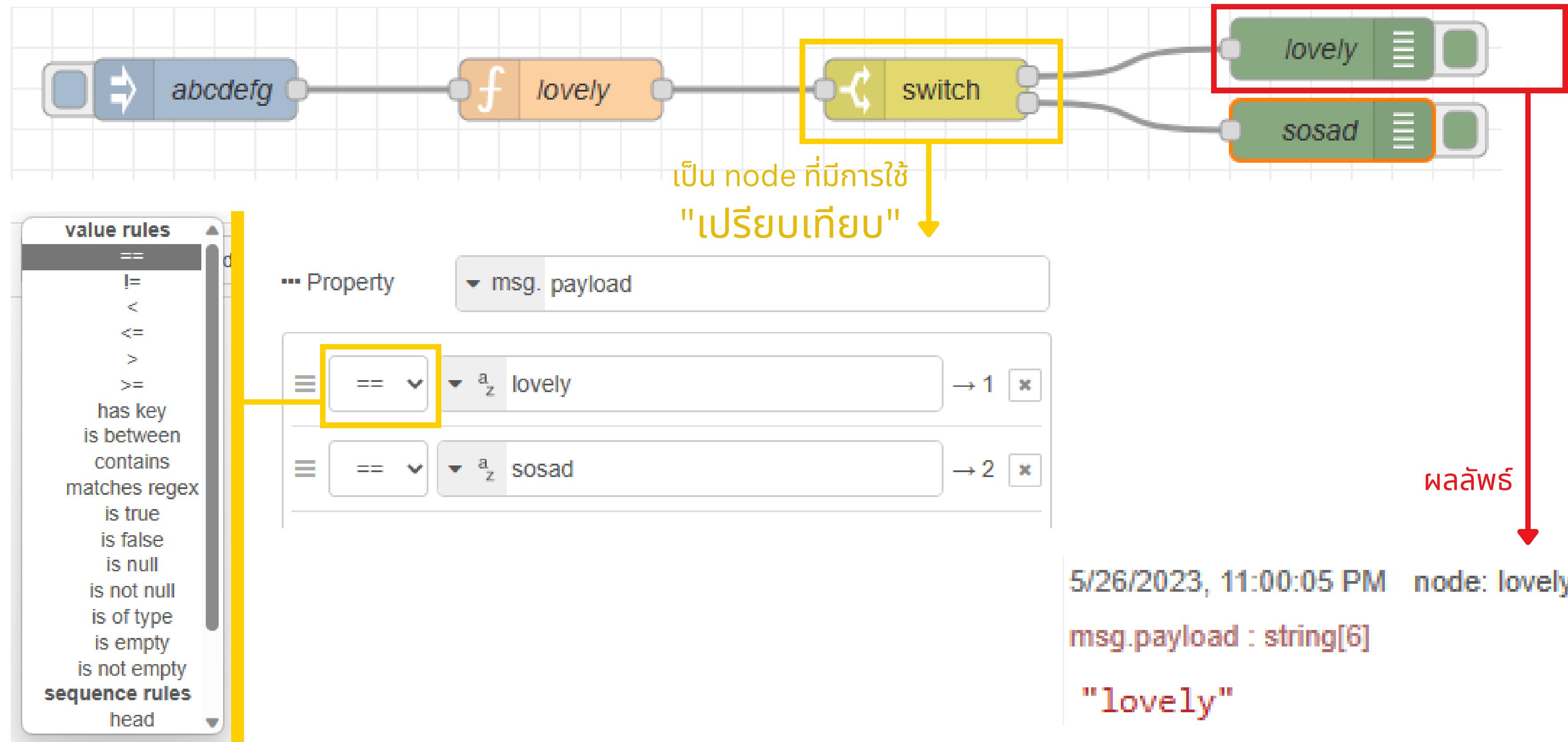
all nodes **all**

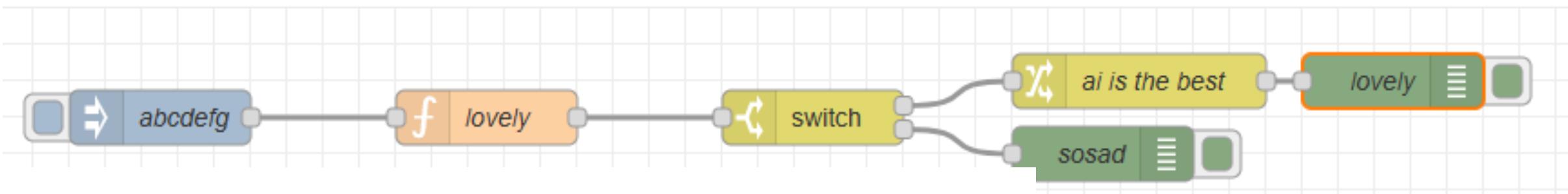
5/27/2023, 11:03:11 AM node: debug 12
msg.payload : string[21]
"first out of output 1"

5/27/2023, 11:03:11 AM node: debug 12
msg.payload : string[22]
"second out of output 1"

5/27/2023, 11:03:11 AM node: debug 12
msg.payload : string[21]
"third out of output 1"

5/27/2023, 11:03:11 AM node: debug 13
msg.payload : string[26]
"only message from output 2"





Rules

Set msg. payload to the value

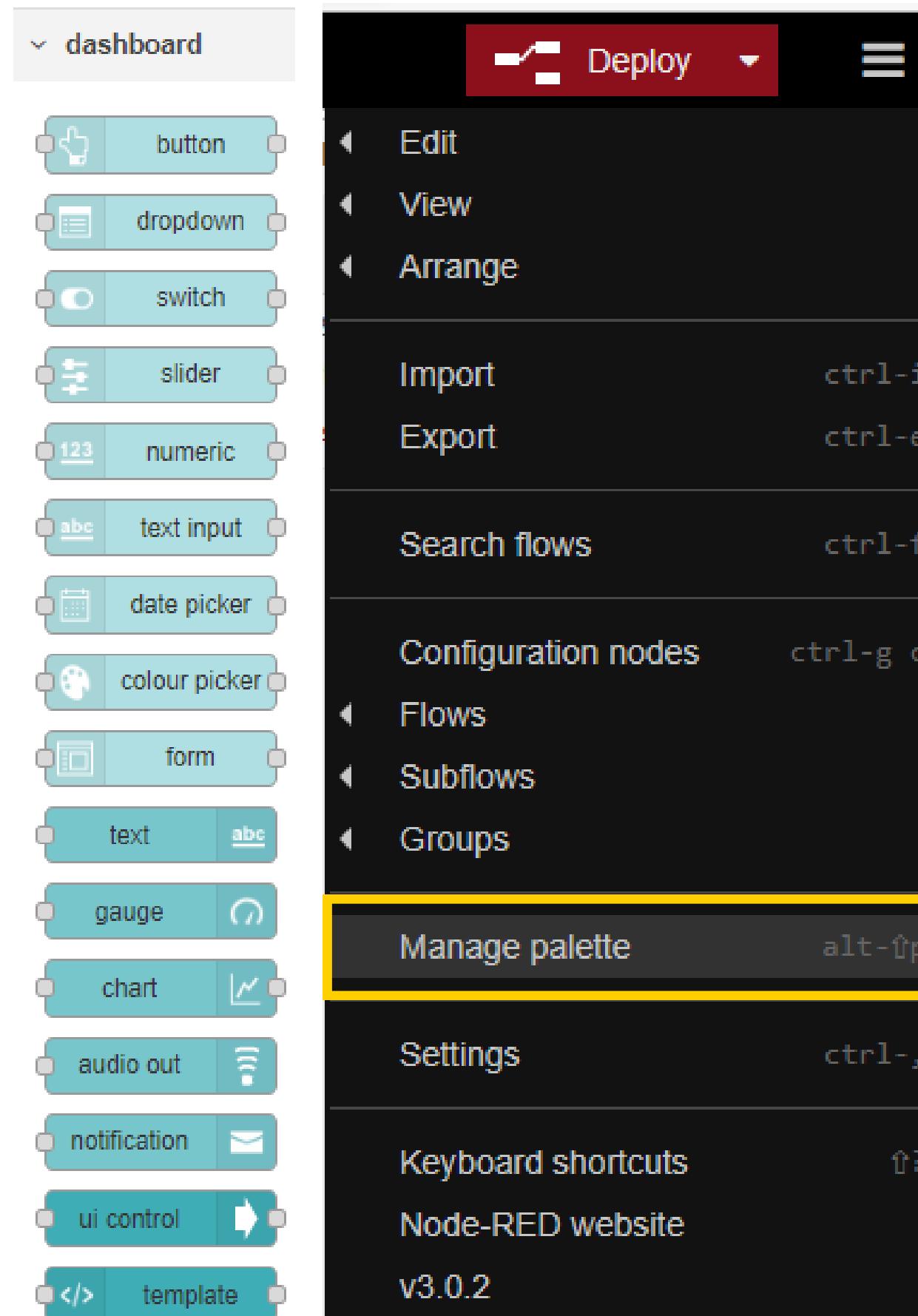
Set

Set
Change
Delete
Move

5/26/2023, 11:02:09 PM node: lovely

msg.payload : string[14]

"ai is the best"

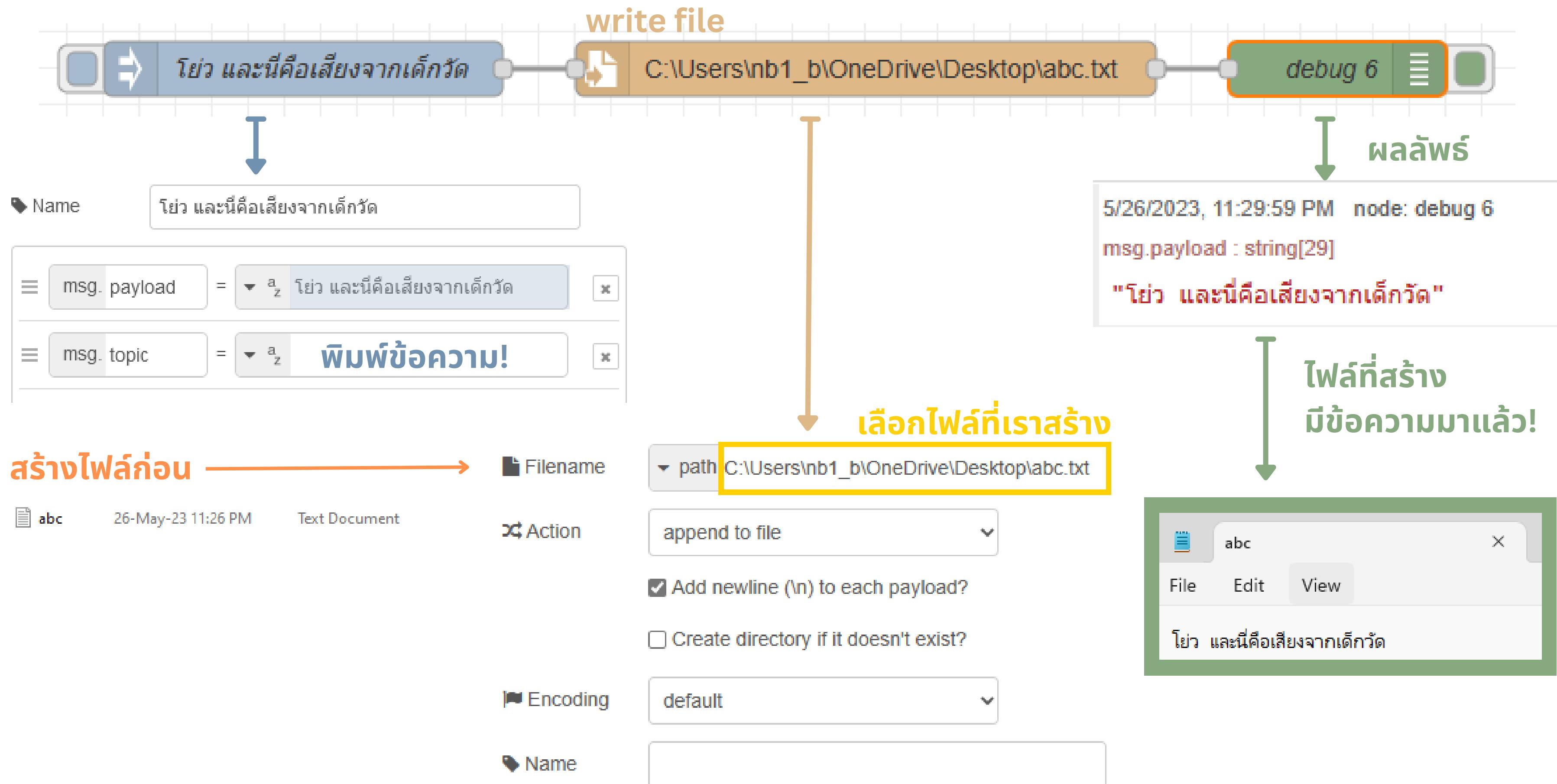


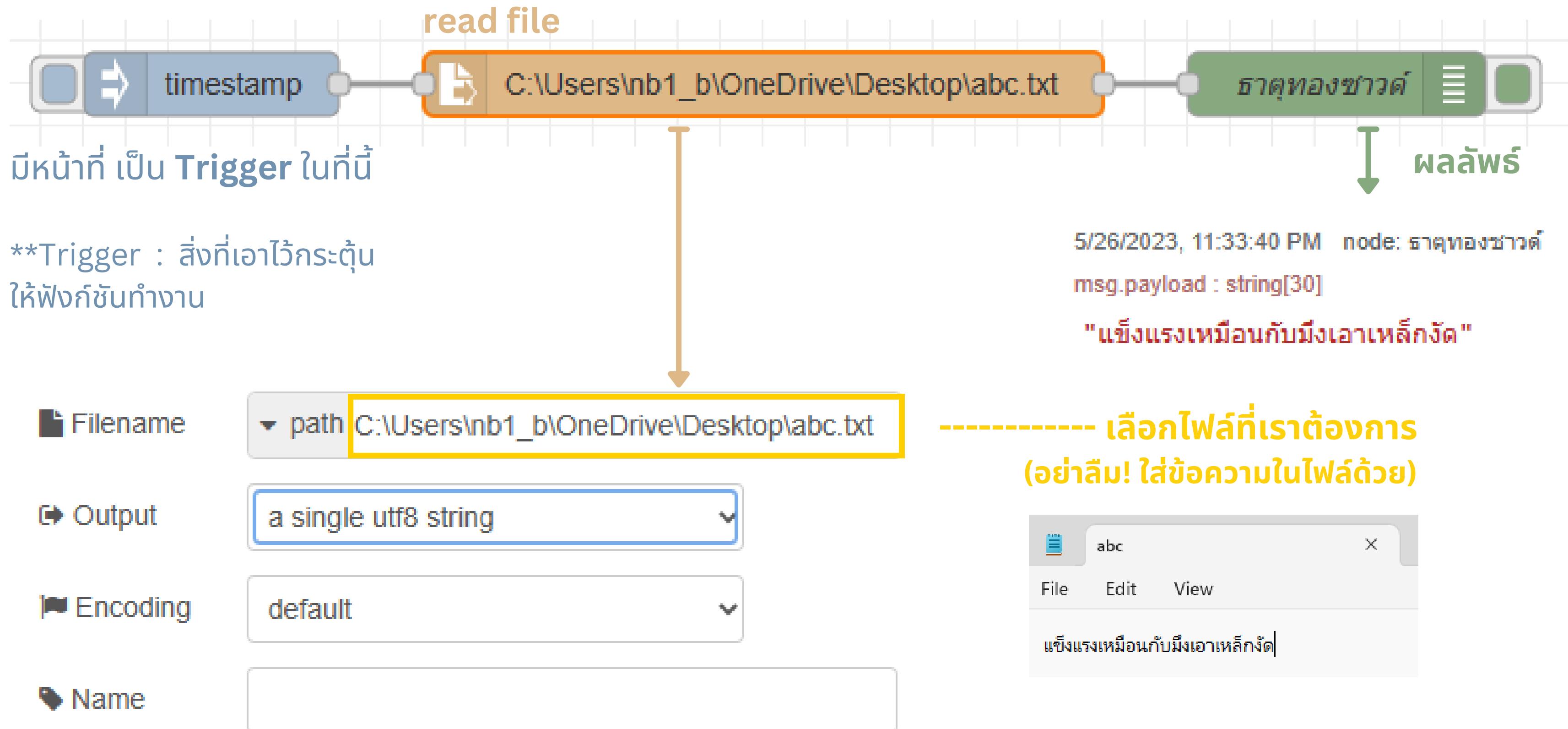
Install <Palette>

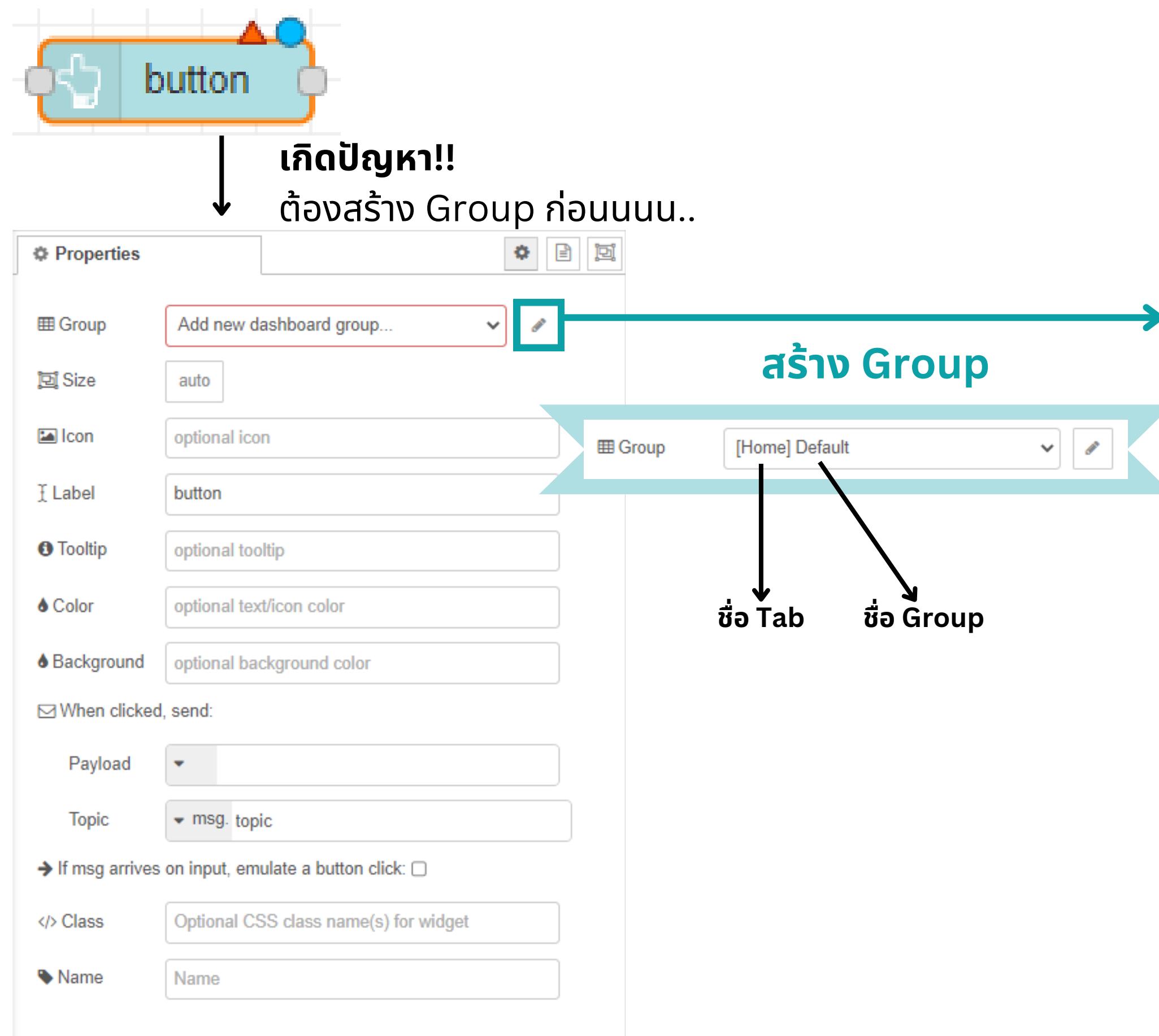
เปิดดูหน้า นี่ จาก... <http://127.0.0.1:1880/ui>

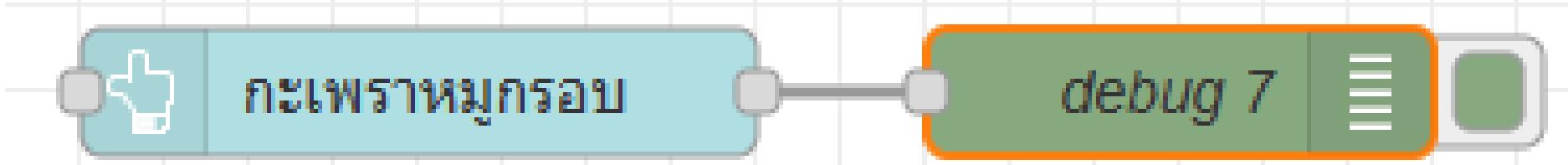
ติดตั้ง
node-red-dashboard

A screenshot of the Node-RED palette management interface. It shows a search bar with "node-red-dashboard" and a results list. The first result is "node-red-dashboard" with version 3.5.0, released 1 day ago. There is a yellow "install" button next to it. The interface includes tabs for "Nodes" and "Install", sorting options (sort: a-z, recent), and a page indicator (17 / 4402).







button

[Home] หัวข่าว สร้างกลุ่มก่อน แล้วมันจะแสดงชื่อที่ตั้งไว้

Icon รูป icon (สามารถใช้ไฟล์จากเครื่องหรือลิงค์ได้)

Label ข้อความบนปุ่ม

Tooltip เมื่อเราวางเมาส์เหนือ Label

Color สีข้อความ

Background สีพื้นหลัง

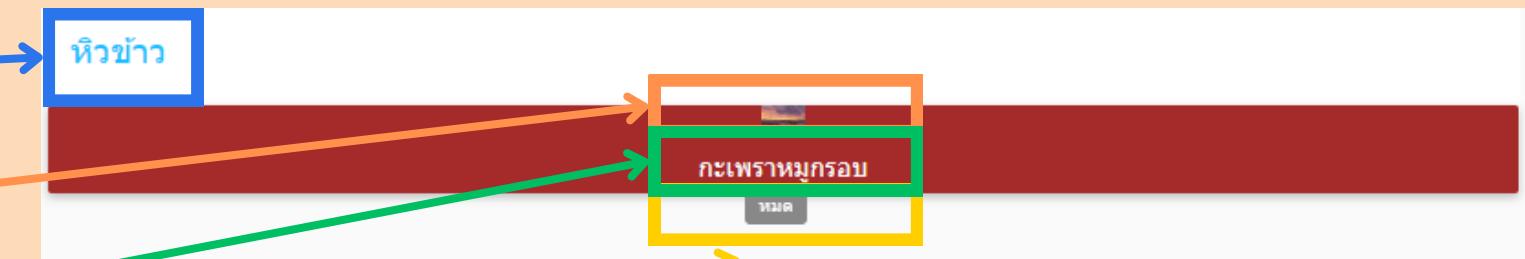
When clicked, send: เมื่อกดคลิก

Payload i'm really hungry ข้อความที่จะส่งไป

Topic msg. topic ตัวแปร หรือ key

If msg arrives on input, emulate a button click:

Name

หน้า UI หลังจากกด Deploy**แลบ Debug**

5/26/2023, 11:38:28 PM node: debug 7

msg.payload : string[17]

"i'm really hungry"

dropdown

The screenshot shows the configuration interface for a dropdown node in Node-RED. The node itself is labeled "กินไวน์ดี". Below it, the "debug 8" node is connected. The configuration panel includes:

- Group:** [Home] หัวข่าว
- Size:** auto
- Label:** กินไวน์ดี
- Tooltip:** ผู้เสีย
- Placeholder:** เมนู ข้อความตัวอย่างหรือคำอธิบาย
- Options:** A list of three items: "ข้าวไข่เจียว" (highlighted with a pink border), "ราดหน้า" (highlighted with a yellow border), and "ไม่กินล่ะ". The text "ค่าในตัวแปรที่ส่งออก" is written over the first item, and "ค่าที่แสดงผ่าน ui" is written over the second item.
- Allow multiple selections from list:**
- If msg arrives on input, pass through to output:**

หน้า UI หลังจากกด Deploy

The dashboard shows the deployed state of the Node-RED application. It includes:

- A header with "กินไวน์ดี" and a "เมนู" tab (highlighted in blue).
- A dropdown menu with three options: "ข้าวไข่เจียว" (selected and highlighted in blue), "ราดหน้า", and "ไม่กินล่ะ". A large blue arrow points downwards from this menu towards the debug log.
- A "Debug" section with the text "กด เลือก".
- A "Debug" log window showing the message: "5/26/2023, 11:41:21 PM node: debug 8 msg.payload : number 2".

switch

```

graph LR
    S1[switch abcd] --> D1[debug 9]
  
```

Configuration details:

- Group: [Home] ห้องข้าว
- Size: auto
- Label: abcd
- Tooltip: optional tooltip
- Icon: Default
- Pass through msg if payload matches valid state:
- When clicked, send:
 - On Payload: abc
 - Off Payload: false
- Topic: msg.topic
- </> Class: Optional CSS class name(s) for widget
- Name:

หน้า UI หลังจากกด Deploy



แลบ Debug

```

5/27/2023, 12:16:05 PM node: debug 9
msg.payload : string[3]
"abc"

5/27/2023, 12:16:06 PM node: debug 9
msg.payload : boolean
false
  
```

slider

```

graph LR
    S[slider hungry level] --> D[debug 10]
  
```

[Home] หัวข้อ

Group

Size

Label

Tooltip

Range min 0 max 700 step 7

Output only on release

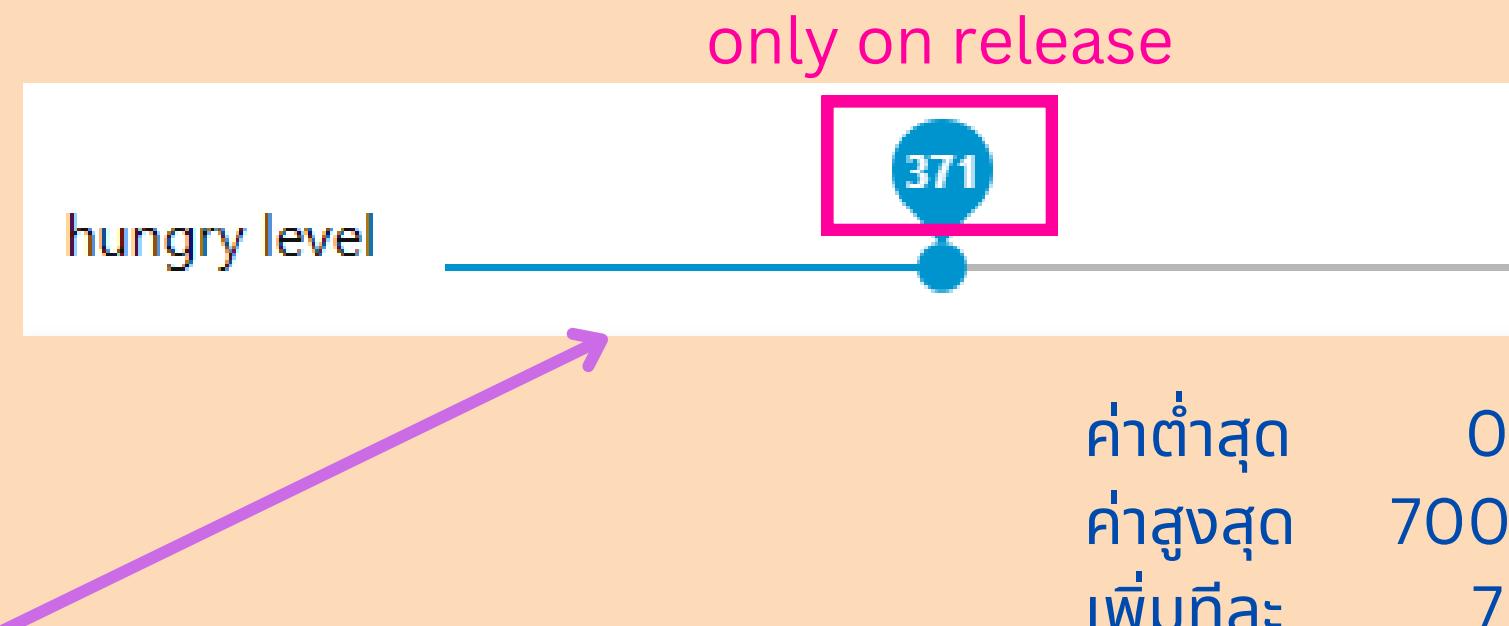
If msg arrives on input, pass through to output:

When changed, send:

Payload Current value

Topic msg. topic

หน้า UI หลังจากกด Deploy



แลบ Debug

5/27/2023, 12:17:36 PM node: debug 10

msg.payload : number

371

numeric

The screenshot shows a Node-RED dashboard with two nodes: a "numeric" input node and a "debug 11" output node. The numeric node has a value of "123". A purple arrow points from the "Range" settings of the numeric node to the "msg.payload" field of the debug node's log entry.

Group: [Home] ห้องข้าว

Size: auto

Label: numeric

Tooltip: noop

Value Format: {{value}}

Range: min 0 max 10 step 1

Wrap value from max to min and min to max:

If msg arrives on input, pass through to output:

When changed, send:

Payload: Current value

Topic: msg.topic

หน้า UI หลังจากกด Deploy

numeric

6

แลบ Debug

5/27/2023, 12:18:56 PM node: debug 11

msg.payload : number

1

5/27/2023, 12:18:57 PM node: debug 11

msg.payload : number

2

text input

```

graph LR
    A[text input] --> B[debug 12]
    
```

Properties:

- Group: [Home] ห้องข้าว
- Size: auto
- Label: yuka
- Tooltip: barry
- Mode: text input
 - text input
 - email address
 - password
 - number
 - telephone input
 - color picker
 - time picker
 - datetime picker
 - week picker
 - month picker
- If msg arrives on topic: to output: checked
- Delay (ms): 300
- Send value on topic:
- When change:
- Payload:
- Topic:

เลือกชนิดของช่องกรอกข้อความ

หน้า UI หลังจากกด Deploy



แลบ Debug

5/27/2023, 12:20:46 PM node: debug 12
msg.payload : string[6]
"lovely"

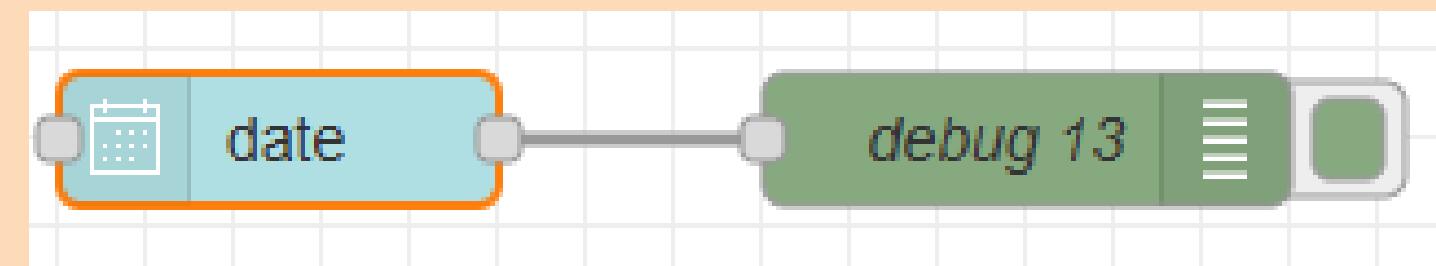
Default

date  28/05/2023

กดเลือกวันที่ได้



date



ແຄບ Debug

5/27/2023, 12:22:01 PM node: debug 13
msg.payload : number
2023-05-27T00:00:00.000Z

感應裝置

debug 14

Group: [Home] ห้องข้าว

Size: auto

Label: สีน้ำเงิน

Format

- hex
- round

Show hue slider:

Show lightness slider:

If width is 4 or greater:

- Always show swatch:
- Always show picker:
- Always show value field:

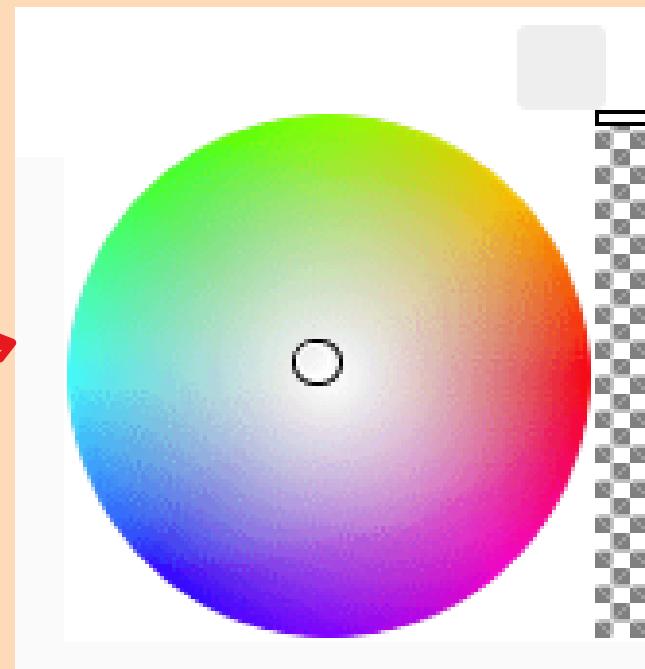
→ If msg arrives on input, pass through to output:

Send: one value when released/closed

Payload: current value as a string

Topic: msg. topic

หน้า UI หลังจากกด Deploy



แลบ Debug

5/27/2023, 12:23:17 PM node: debug 14

msg.payload : string[6]

"f90606"

The screenshot shows a Node-RED flow. A 'justform' node is connected to a 'debug 15' node. Below the flow, there is a configuration panel for the 'justform' node.

Form elements:

Label	Name	Type	Required
name	jsonkey	Text	On
num	a	Number	Off
mail	b	E-mail	Off
time	c	Date	Off

Buttons:

- submit
- cancel

Topics:

- msg. topic

Labels:

- [Home] หัวข้อ
- auto
- justform

Text:

สามารถเพิ่มได้เรื่อยๆ

ข้อความที่แสดง ชื่อตัวแปรที่จะเอา ชนิดข้อความ **บังคับกรอกใหม่?
ค่าใส่

หน้า UI หลังจากกด Deploy

The screenshot shows the deployed Node-RED dashboard. The 'justform' configuration is displayed with sample data:

- name: Poomphussakorn Poomee
- num: 123546789765432
- mail: abcd@gmail.com
- time: 27-May-2023

Below the form, there are 'SUBMIT' and 'CANCEL' buttons.

ผล Debug

5/27/2023, 12:25:56 PM node: debug 15

msg.payload : Object

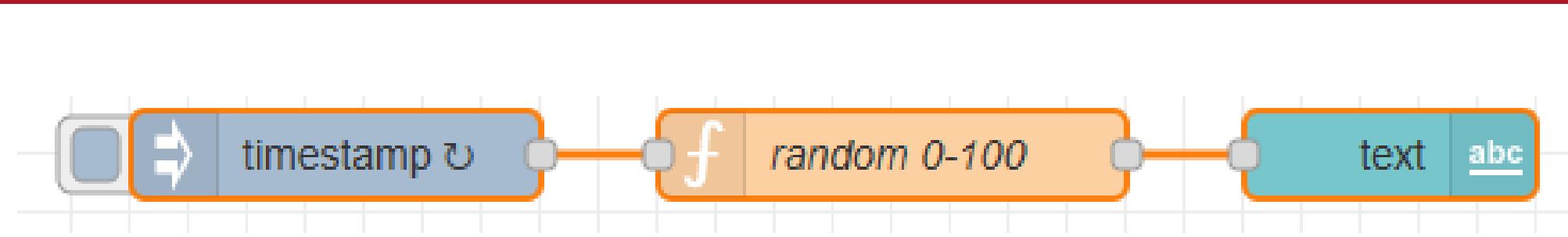
object

jsonkey: "Poomphussakorn Poomee"

a: 123546789765432

b: "abcd@gmail.com"

c: "2023-05-26T17:00:00.000Z"



■ Group [Home] ห้องข้าว ▼

■ Size auto

■ Label text

■ Value format {{msg.payload}}

■ Layout

label value	label value	label value
label value	label value	

</> Class Optional CSS class name(s) for widget

>Name

หน้า UI หลังจากกด Deploy

text 53

The screenshot shows a Node-RED dashboard with a flow and a configuration panel for a gauge.

Flow:

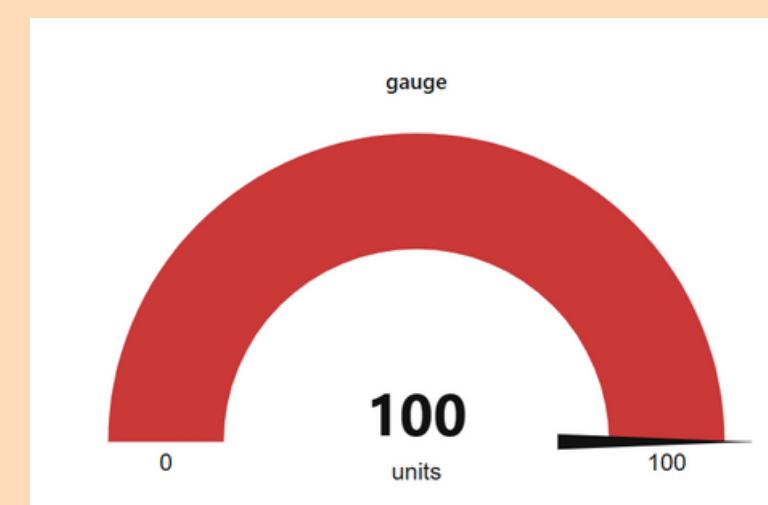
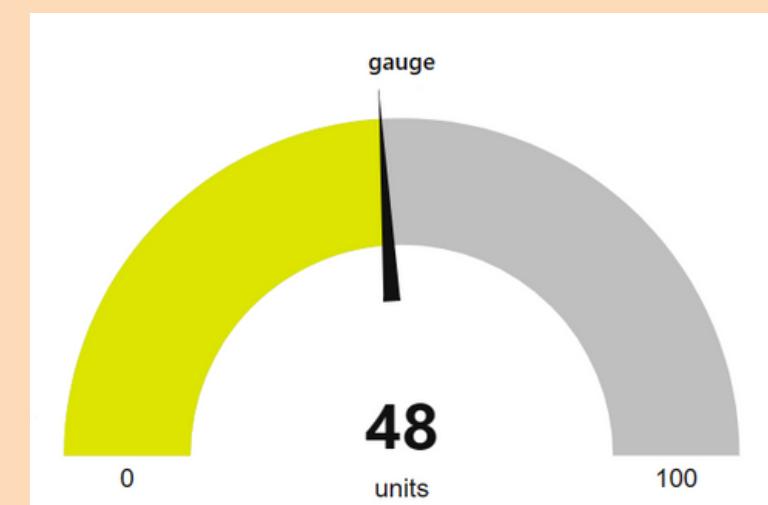
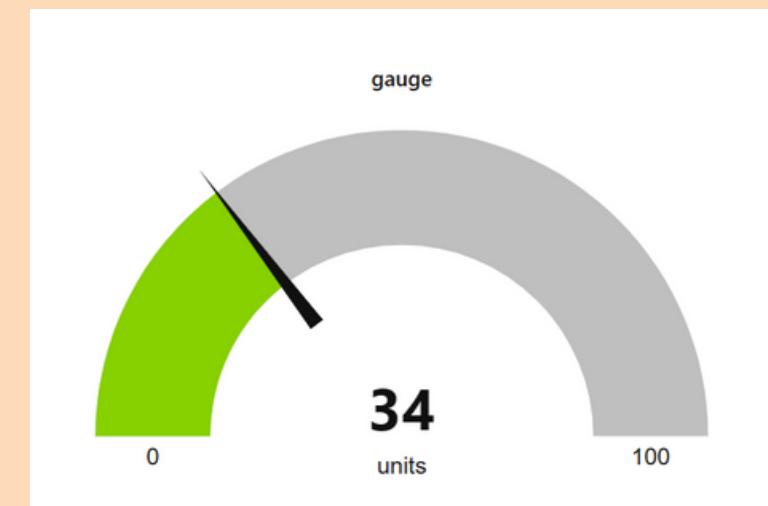
```
graph LR; timestamp((timestamp)) --> randomNode[random 0-100]; randomNode --> textNode[text abc]; textNode --> gaugeNode[gauge]
```

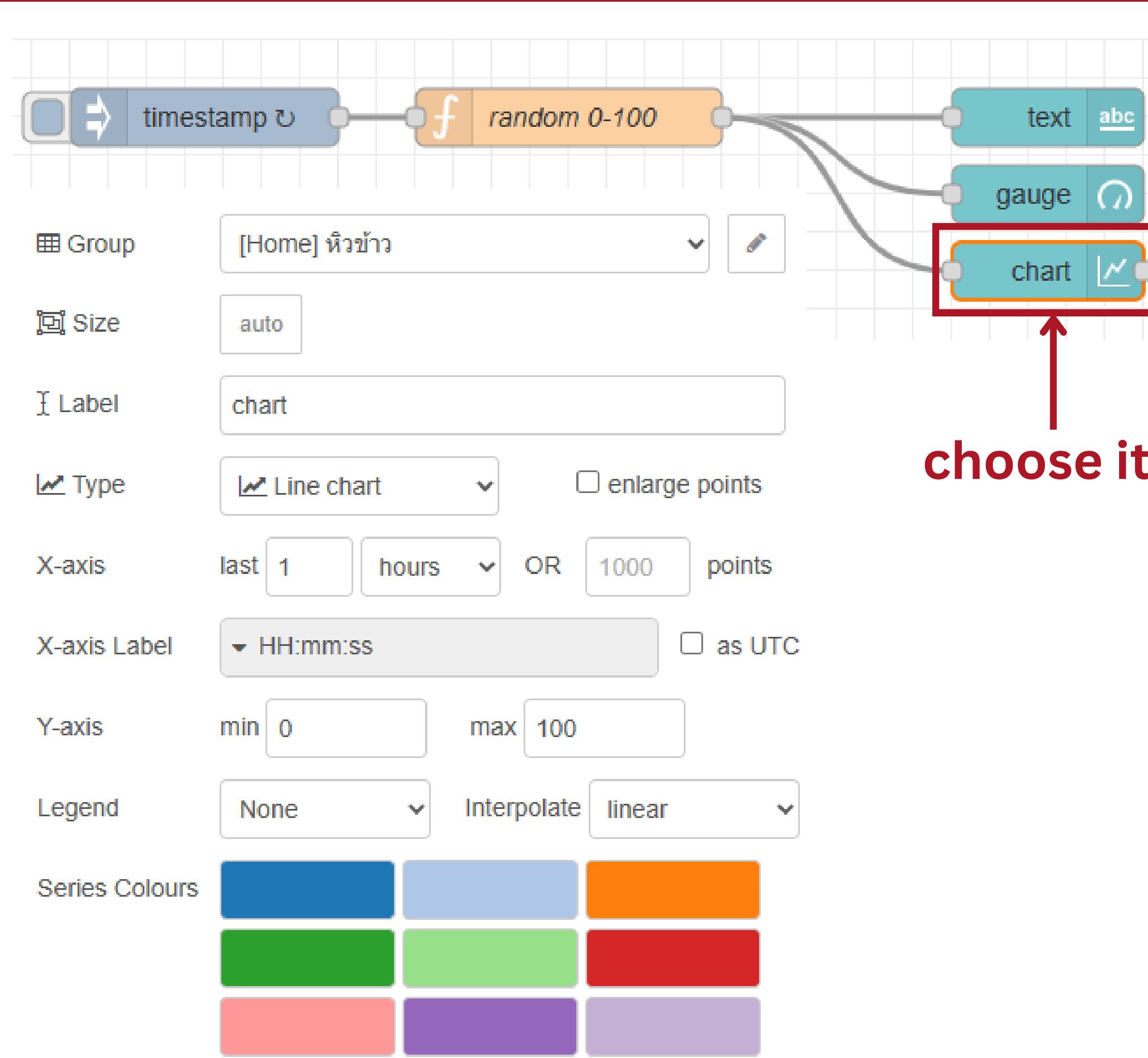
Configuration Panel:

- Group: [Home] ห้องข้าว
- Size: auto
- Type: Gauge
- Label: gauge
- Value format: {{value}}
- Units: units
- Range: min 0 max 100
- Colour gradient: A gradient bar with three segments: green, yellow, and red.
- Sectors: 0 ... optional ... optional ... 10
- Fill gauge from centre:
- </> Class: Optional CSS class name(s) for widget
- Name: (empty input field)

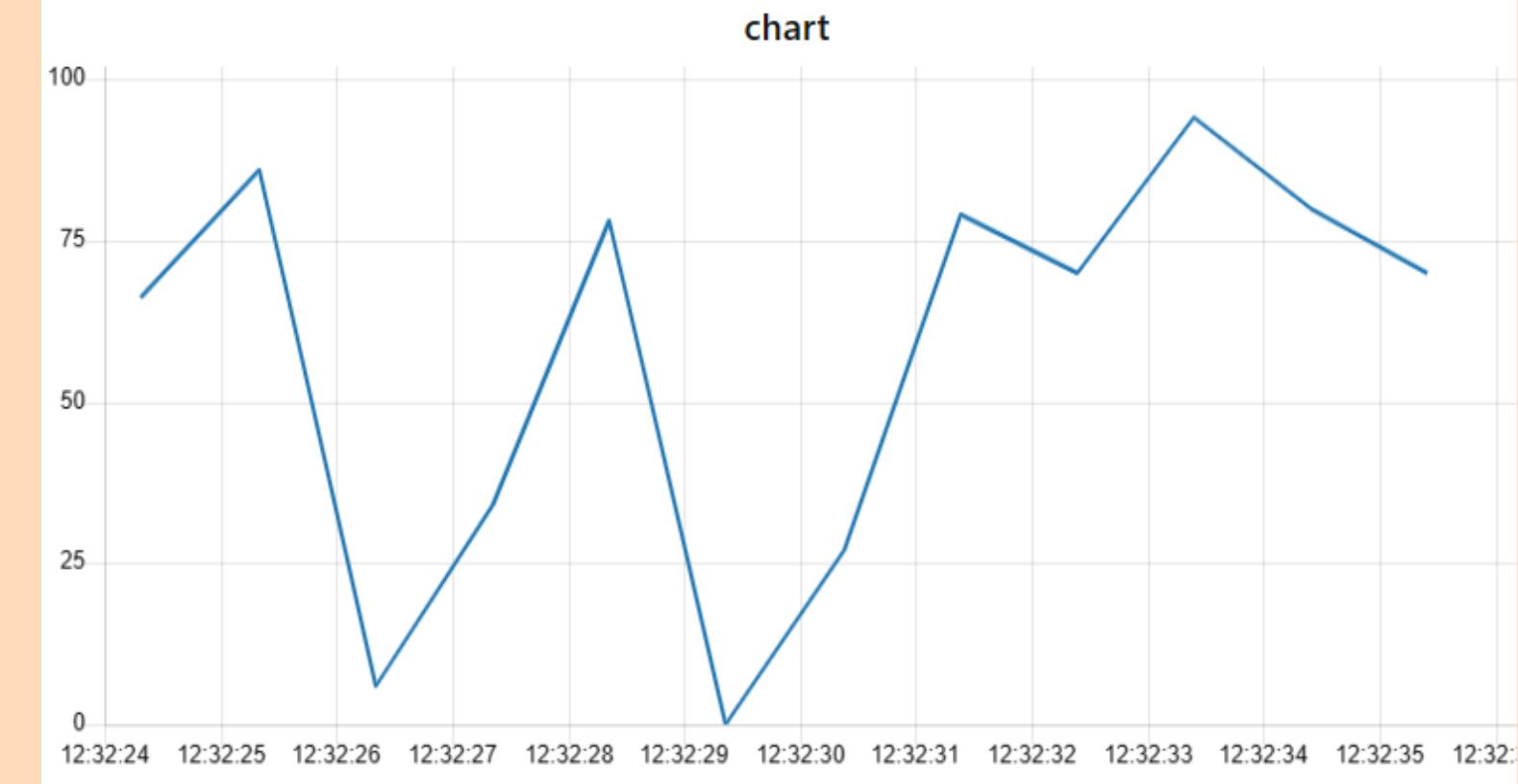
A red arrow points from the "choose it!" text to the gauge node in the flow.

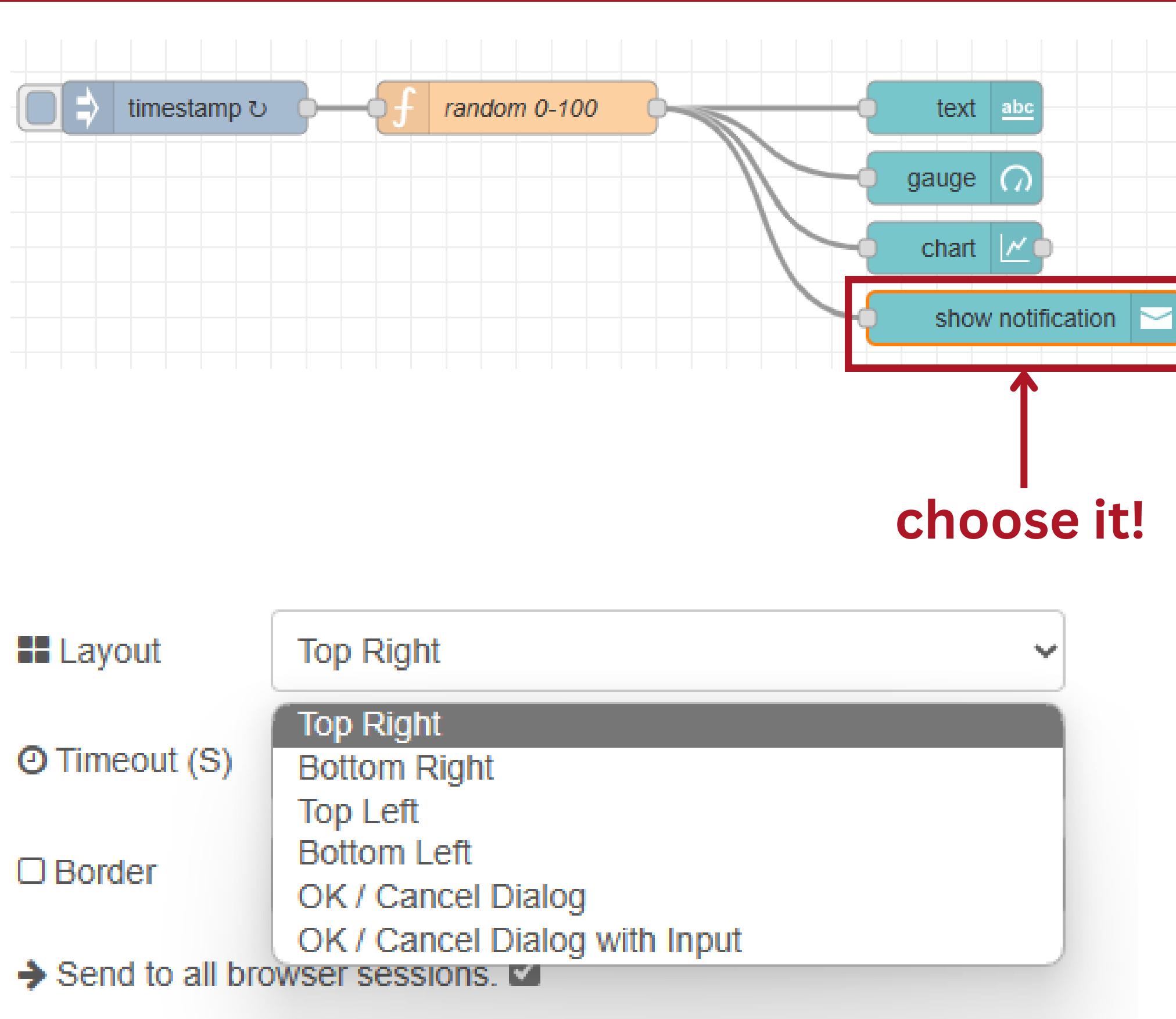
หน้า UI หลังจากกด Deploy





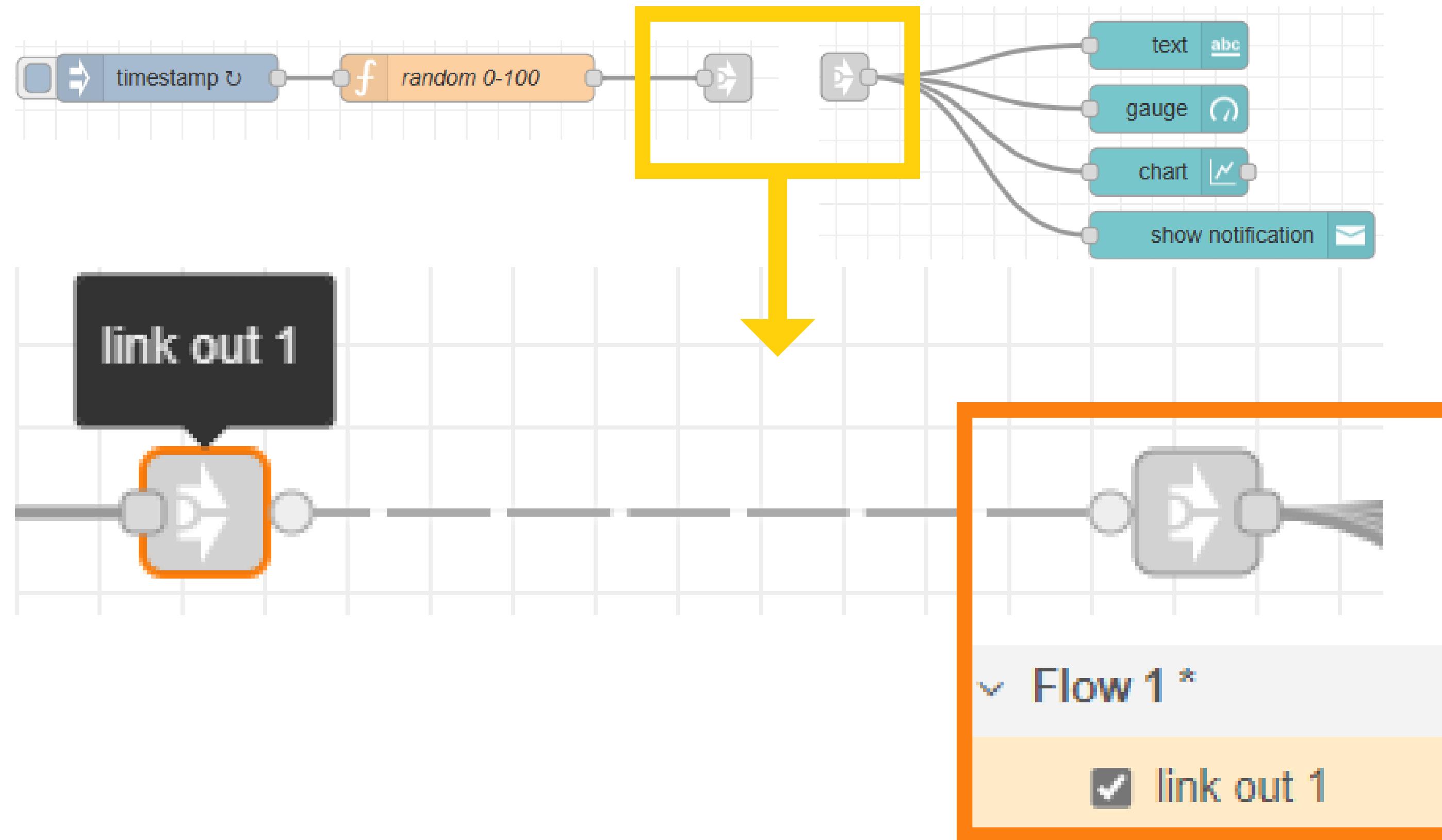
หน้า UI หลังจากกด Deploy





หน้า UI หลังจากกด Deploy

choose it!



	TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)
ข้อดี	มีการรับประกันในการส่งข้อมูล, มีการตรวจสอบข้อผิดพลาด และการควบคุมการให้ลบทองข้อมูล, จึง เหมาะสมกับการส่งข้อมูลที่ต้องการความน่าเชื่อถือสูง	การส่งข้อมูลเร็วและง่าย ไม่ต้องสร้างการเชื่อมต่อ ก่อนส่งข้อมูล, เหมาะสมสำหรับการส่งข้อมูลที่ไม่จำเป็นต้องมีการรับประกัน
ข้อเสีย	มีการส่งข้อมูลที่ช้ากว่า UDP เนื่องจากต้องมีการสร้างการเชื่อมต่อและการยืนยันข้อมูล	ไม่มีการรับประกันในการส่งข้อมูล, ไม่มีการตรวจสอบข้อผิดพลาดหรือการควบคุมการให้ลบทองข้อมูล



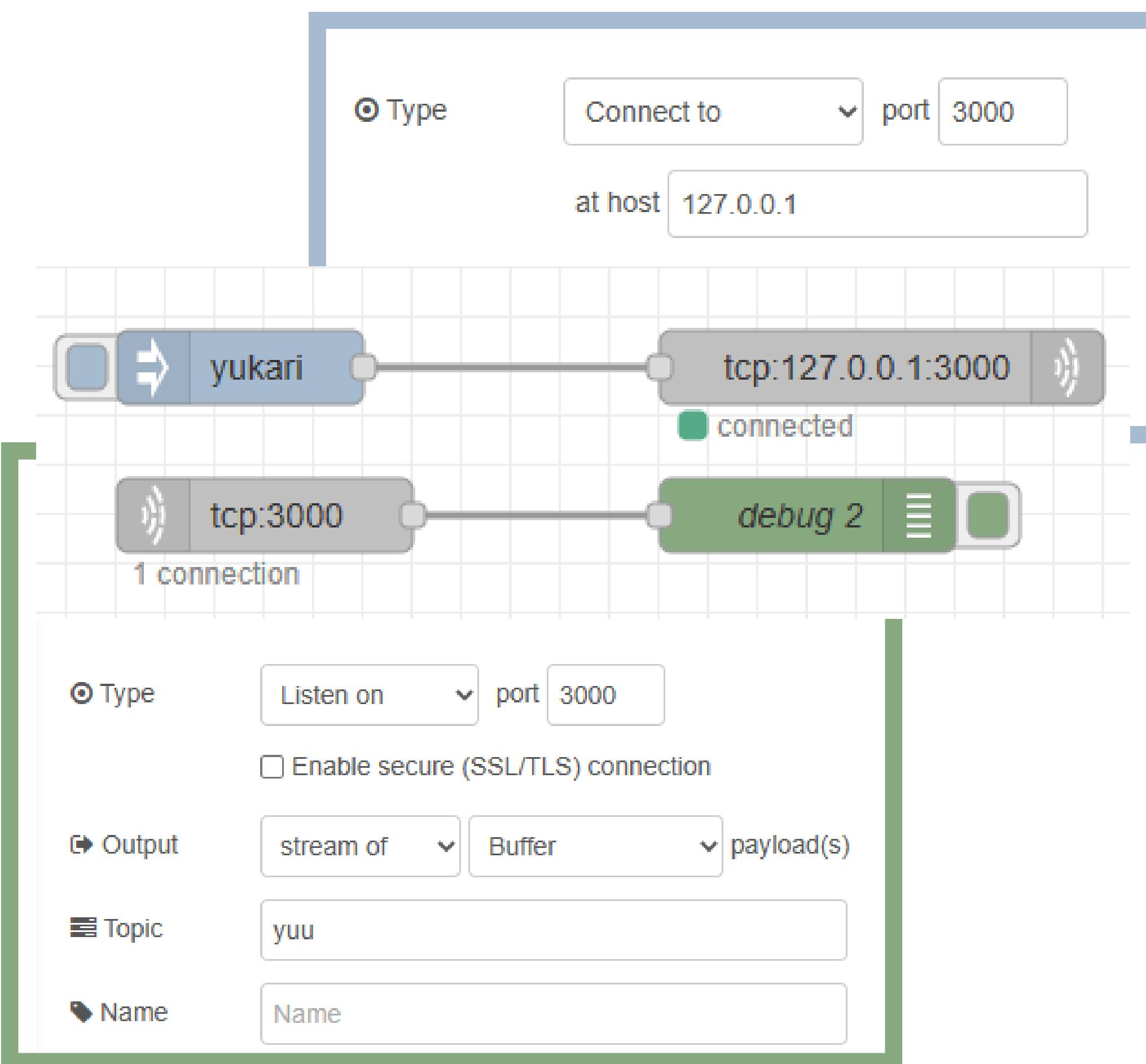
ແລ້ວ Debug

5/26/2023, 11:17:06 PM node: debug 3

msg.payload : buffer[5]

▼ buffer[5] string

barry



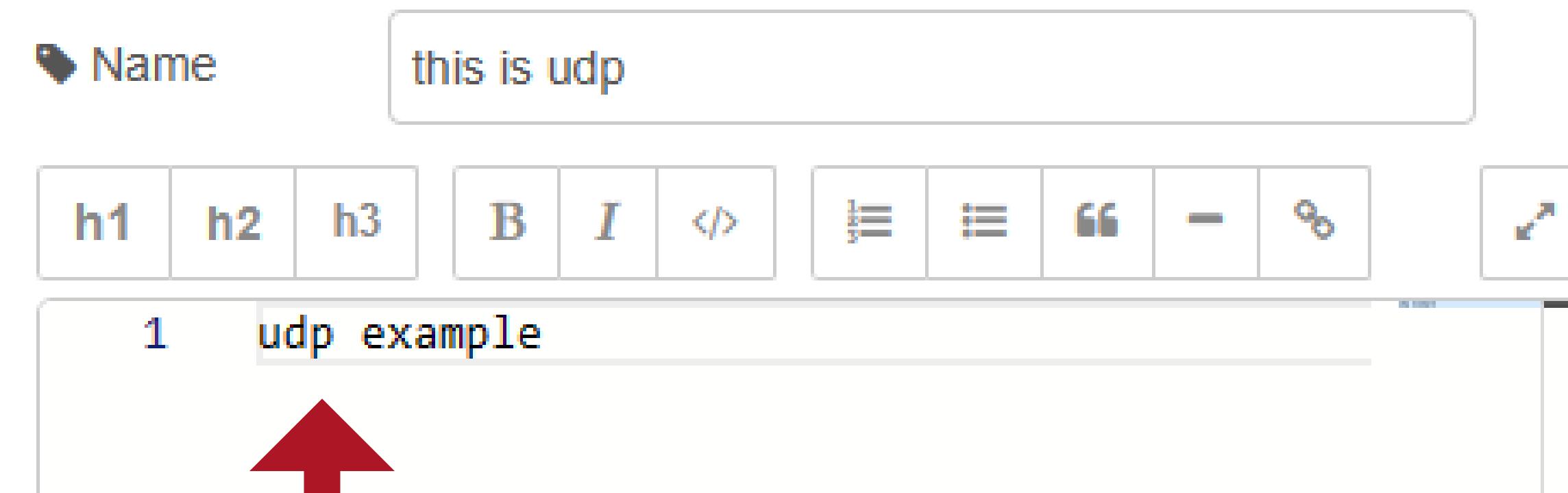
ແກບ Debug

5/26/2023, 11:15:05 PM node: debug 2

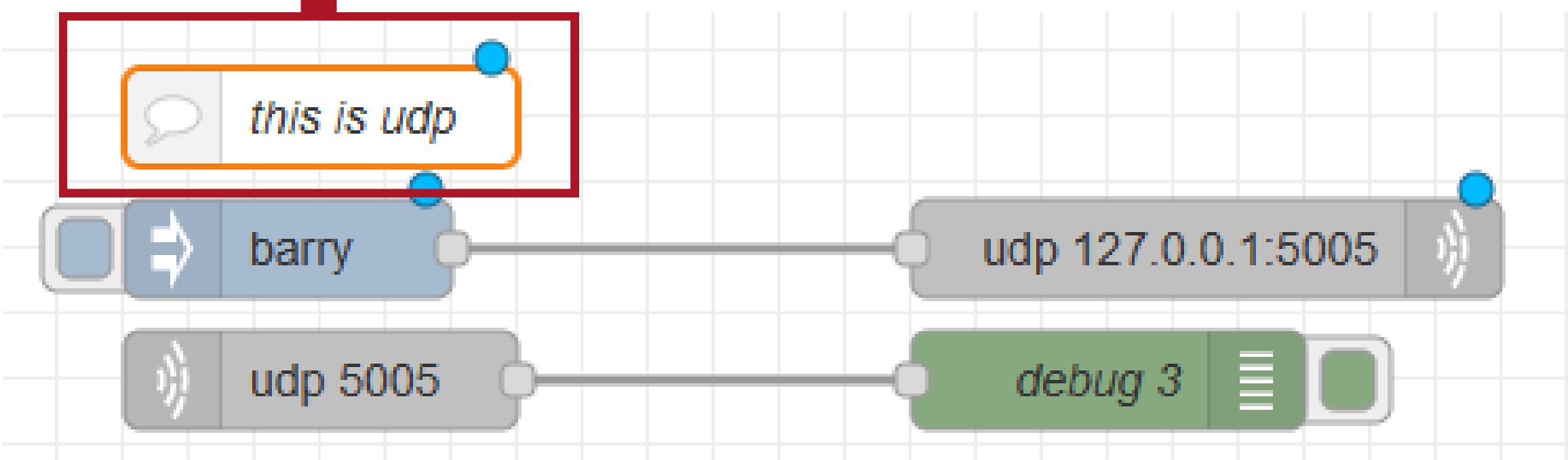
yuu : msg.payload : buffer[6]

▼ buffer[6] string

yukari



เป็นการเขียนข้อความอธิบายการทำงานต่างๆ
ซึ่ง node red จะไม่อ่านในส่วนนี้

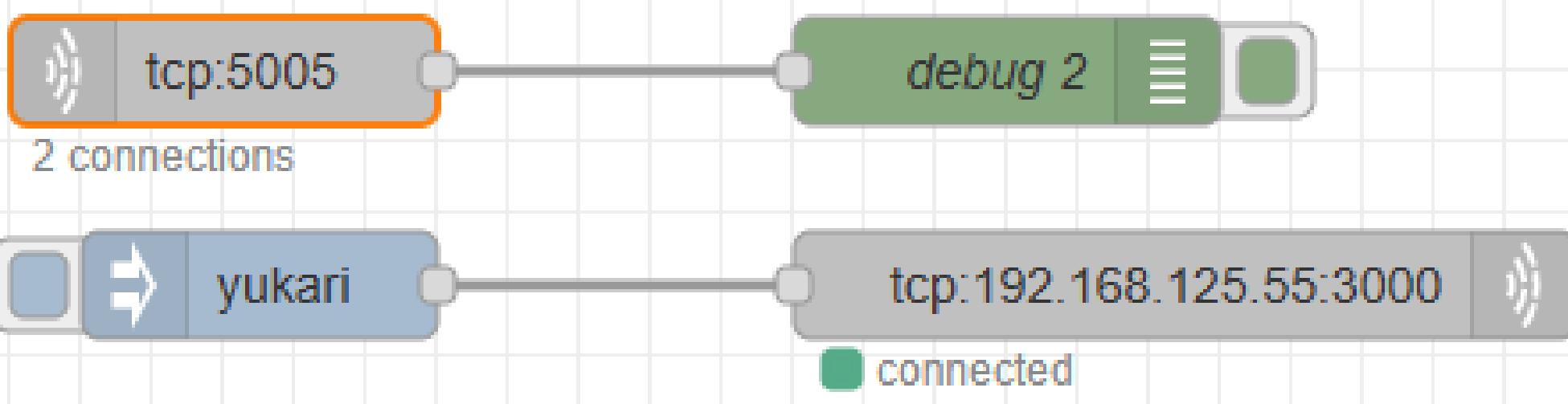


TCP SEND AND RECEIVE DATA

```
1 #include <WiFi.h>
2 #include <WiFiClient.h>
3 #include <WiFiServer.h>
4
5 const char* ssid = "your_SSID";
6 const char* password = "your_PASSWORD";
7 const char* serverIP = "node-red_IP_address";
8 const int serverPort = 5005;
9 const int clientPort = 3000;
10
11 WiFiClient client; // TCP client
12 WiFiServer server(clientPort); // TCP server listening on port 3000
13
14 void setup() {
15     Serial.begin(115200);
16
17     WiFi.begin(ssid, password);
18     while (WiFi.status() != WL_CONNECTED) {
19         delay(1000);
20         Serial.println("Connecting to WiFi...");
21     }
22     Serial.println("Connected to WiFi");
23
24     server.begin(); // Start the TCP server
25     Serial.println("TCP server started");
26 }
27
```

```
ho 0 tail 12 room 4
load:0x40078000,len:10944
load:0x40080400,len:6388
entry 0x400806b4
Connecting to WiFi...
Connecting to WiFi...
Connecting to WiFi...
Connected to WiFi
TCP server started
Connected to Node-RED server
New client connected
Received message from client: yukari
```

```
28 void loop() {
29     // Check for incoming client connections
30     WiFiClient clientConnection = server.available();
31     if (clientConnection) {
32         Serial.println("New client connected");
33         while (clientConnection.connected()) {
34             if (clientConnection.available()) {
35                 String message = clientConnection.readStringUntil('\n');
36                 Serial.println("Received message from client: " + message);
37
38                 // Process the received message from the client as needed
39
40                 // Reply to the client
41                 clientConnection.println("Message received by server");
42                 clientConnection.flush();
43             }
44         }
45         // Client disconnected
46         Serial.println("Client disconnected");
47         clientConnection.stop();
48     }
49
50     // Check if TCP client is connected and send a message
51     if (client.connected()) {
52         client.println("THIS IS LORD BARRY");
53         client.flush();
54         delay(1000); // Wait for a response from the server
55         while (client.available()) {
56             String response = client.readStringUntil('\n');
57             Serial.println("Received response from server: " + response);
58         }
59     } else {
60         // Attempt to connect to the Node-RED server
61         if (client.connect(serverIP, serverPort)) { // Replace with your Node-RED server IP and port
62             Serial.println("Connected to Node-RED server");
63         }
64     }
65
66     // Other code or tasks can be added here
67 }
```



Manage devices

Manage allowed devices

Device limit

Limit number of devices to improve performance

Connected devices

NB-RD-29-09-22-30

IP address:192.168.125.105/24

MAC address:84:7B:57:67:5D:4F

esp32-arduino

IP address:192.168.125.55/24

MAC address:4C:11:AE:EF:51:EC

5/27/2023, 5:19:32 PM node: debug 2

yuu : msg : Object

* object

topic: "yuu"

* payload: buffer[18] string

THIS IS LORD BARRY

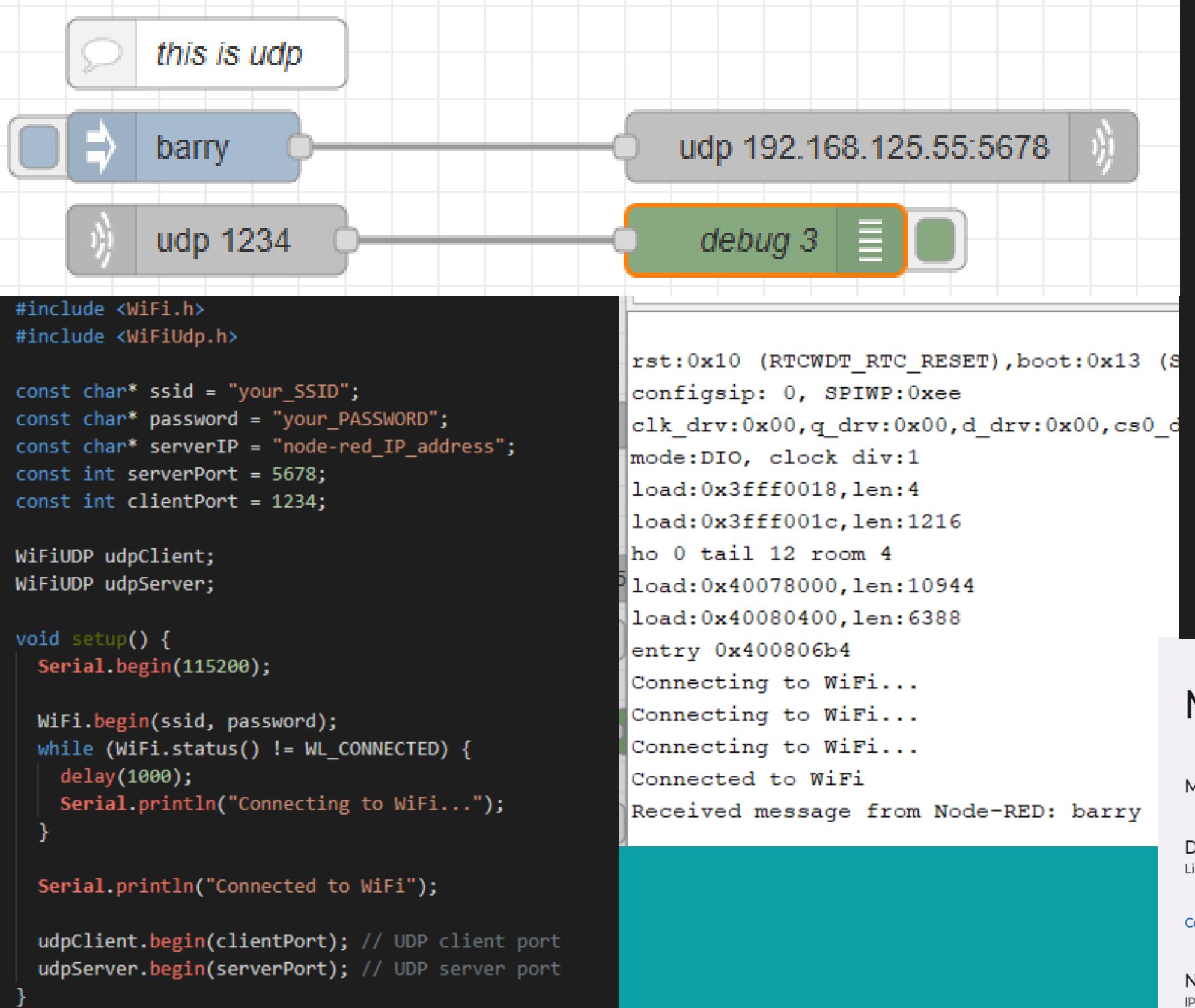
ip: "::ffff:192.168.125.55"

port: 49808

* _session: object

_msgid: "6141cf513d9f8b29"

UDP SEND AND RECEIVE DATA



```
28 void loop() {
29   // UDP Client (send message to Node-RED server)
30   udpClient.beginPacket(serverIP, clientPort); // Node-RED server IP and port
31   udpClient.println("YUKARI");
32   udpClient.endPacket();
33
34   // UDP Server (receive message from Node-RED server)
35   int packetSize = udpServer.parsePacket();
36   if (packetSize) {
37     char incomingMessage[255];
38     int len = udpServer.read(incomingMessage, 255);
39     if (len > 0) {
40       incomingMessage[len] = '\0';
41       Serial.print("Received message from Node-RED: ");
42       Serial.println(incomingMessage);
43     }
44   }
45   delay(1000);
46 }
```

5/27/2023, 5:34:58 PM node: debug 3

msg : Object

object

payload: buffer[8] string

YUKARI

Manage devices

Manage allowed devices

Device limit

Limit number of devices to improve performance

Connected devices

NB-RD-29-09-22-30

IP address:192.168.125.105/24

MAC address:84:7B:57:67:5D:4F

esp32-arduino

IP address:192.168.125.55/24

MAC address:4C:11:AE:EF:51:EC

fromip: "192.168.125.55:1234"

ip: "192.168.125.55"

port: 1234

_msgid: "92a4b2f835fd3fee"

GO CREATE

PROJECT

