

物聯網實務

(九)

廖裕評

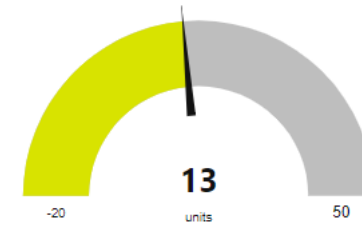
Building a weather station

- Weather APIs give detailed information beyond just temperature, precipitation and humidity, but also UV index and pollution levels. Time series data lets you discover trends and patterns, and plan events based on weather conditions. APIs can also be a useful resource for disaster management as some come with built-in alerting and notification mechanisms on a global scale.

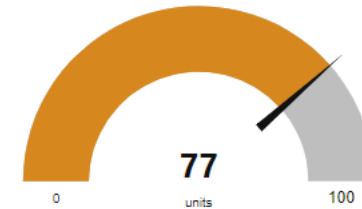
Default

Paris

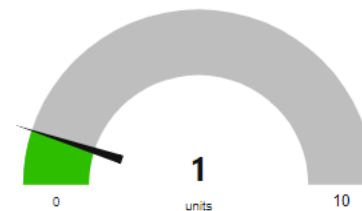
Temperature



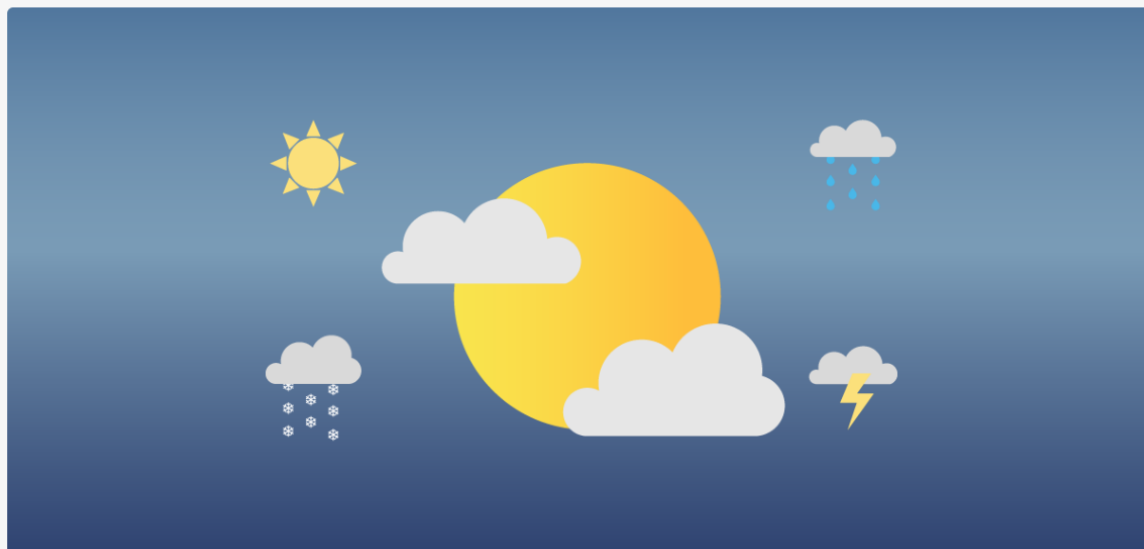
Humidity



UV




8 Best Free and Paid Weather APIs



8 Best Free and Paid Weather APIs

<https://nordicapis.com/6-best-free-and-paid-weather-apis/>

Weather API

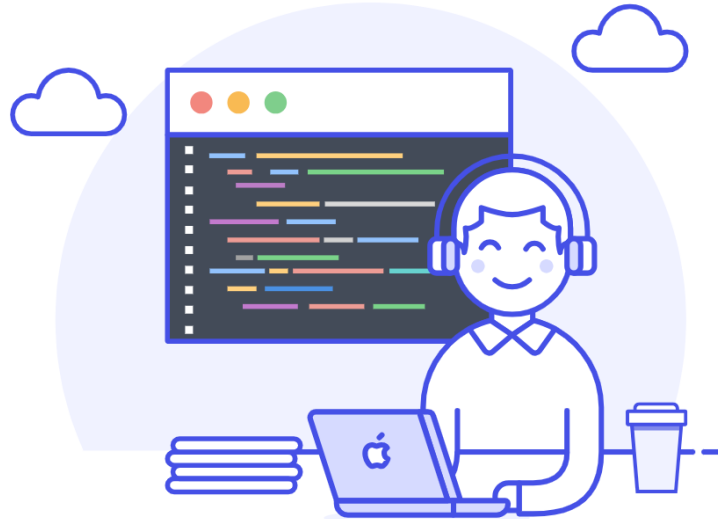
The logo for Weather API, featuring a green cloud with a white drop inside, followed by the text "weather" in a bold sans-serif font and "api" in a smaller, lighter sans-serif font below it.

[Features](#) [Pricing](#) [API Explorer](#) [Docs](#) [Weather](#) [Contact](#) [Signup](#) [Login](#)

JSON and XML Weather API and Geolocation Developer API

Designed for developers by developers, Weather API is the ultimate weather and geolocation API trusted by +310K users worldwide. Integrate weather in [Browser Extension](#).

[Sign Up](#) [View Docs](#)

A line-art illustration of a developer. The developer is wearing a headset and is seated at a desk. On the desk is a laptop with an Apple logo, a stack of books, and a coffee cup. In the background, there is a large window or screen displaying lines of code with various colors for syntax highlighting. The entire scene is set against a light blue circular backdrop, with a few clouds floating around.

<https://www.weatherapi.com/>

Sign Up

On sign up you will be subscribed to Pro Plus plan for free, no obligation 14 day trial. After trial has ended you will be automatically switched to Free plan.

Note: All the fields are required!

Email (Your email is your username)

lyp@cycu.org.tw

Retype Email

lyp@cycu.org.tw

Password

Retype Password

Passwords do not match

Are you human?



I'm not a robot



The verification has failed. Please try again!

☒ I have read and agree to [T&C's](#) and [Privacy Policy](#).

Sign up

Account Activation - WeatherAPI.com

外部

收件匣 x



WeatherAPI.com Support <support@weatherapi.com>

下午6:27 (28 分鐘前)



寄給 我 ▾



英文 ▾



中文 (繁體) ▾

翻譯郵件

關閉下列語言的翻譯功能：英文 x

Hello,

Thanks for signing up.

You will only need to visit the link once to verify and activate your account.

To complete your account verification, please click the link given below:-

<https://www.weatherapi.com/confirm.aspx?code=3191bfab-dc3f-4c89-9e0d-fe401c3e1db0>

If the above link does not work, please copy and paste link into your web browser.

If you are still having problems signing up then please do get in touch.

Thanks,

Sana

weatherapi

Dashboards

API Response Fields

Analytics

Accounts

Change Plan

Payment Method

Billing

Support

API Explorer

Swagger Tool

Change Password

Log out

Welcome Back

API Key: 3191411

Copy

LIVE

TRIAL Ends on 28/Nov/2023

Pro Plus

Plan

5,000,000

Calls per Month

0

Calls Made

28/Nov/23

Trial End Date

Note: If you are on a trial plan then after the trial plan ends your API key will be automatically moved to Free plan if you do not wish to upgrade to a paid plan.

☆ Get Started

• Learn how to form HTTP request to get weather from [API Explorer](#) or use our NEW [Swagger](#)



1

Welcome Back

Dashboards

API Response Fields

Analytics

Accounts

Change Plan

Payment Method

Billing

Support

API Explorer

Swagger Tool

Change Password

Log out

Change Fields

By default we return all weather fields in response and that may not be ideal for your application. So provided below you could enable or disable the fields you want to remove from your response.

If you wish any fields to not appear in the json response then simply uncheck the fields below and click on Save button at the bottom of screen.

API Key: 31 111

2.copy

Current Weather

- ☒ last_updated_epoch
- ☒ last_updated
- ☒ temp_c
- ☒ temp_f
- ☒ is_day
- ☒ text
- ☒ icon
- ☒ code

Marine Weather

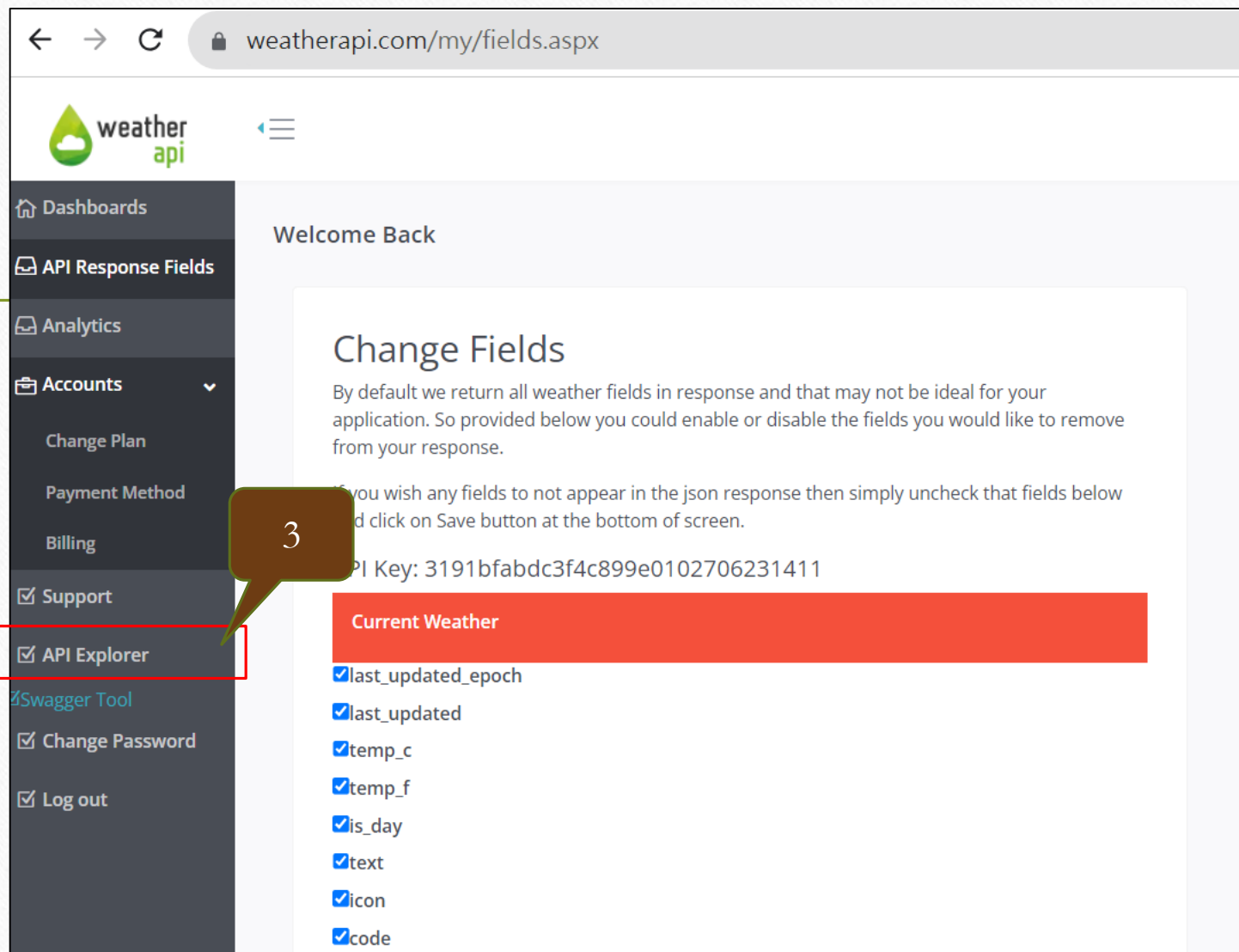
tides

- ☒ tide_time
- ☒ tide_height_mt
- ☒ tide_type

Hour

- ☒ sig_ht_mt
- ☒ swell_ht_mt
- ☒ swell_ht_ft
- ☒ swell_dir
- ☒ swell_dir_16_point
- ☒ swell_period_secs
- ☒ water_temp_c
- ☒ water_temp_f

Save



Interactive API Explorer

WeatherAPI.com interactive API explorer or IO Docs allows you to test our APIs and methods. It returns response headers, response code and response body.

For complete documentation please visit our [Weather API Documentation](#) section.

You can also now use our Swagger tool to learn how to form weather Request. Visit [Swagger Tool](#)

1.Paste

Your API Key

3191bfabdc3f4c899e231411

2.HTTPS

Protocol

HTTPS

3.JSON

Format

JSON

Parameter	Value	Type	Location	Description
q	London	string	query	Pass US Zipcode, UK Postcode, Canada Postalcode, IP address, Latitude/Longitude (decimal degree) or city name. Visit request parameter section to learn more.

Current

Forecast

Search/Autocomplete

History

Future

Marine

Astronomy

Time Zone

Sports

Parameter	Value	Type	Location	Description
aqi	yes	string	query	Get air quality data

5

Show Response

Copy

Call

```
https://api.weatherapi.com/v1/current.json?key=3191bfabdc3f4c899e0102706231411&q=London&aqi=yes
```

Response Code

200

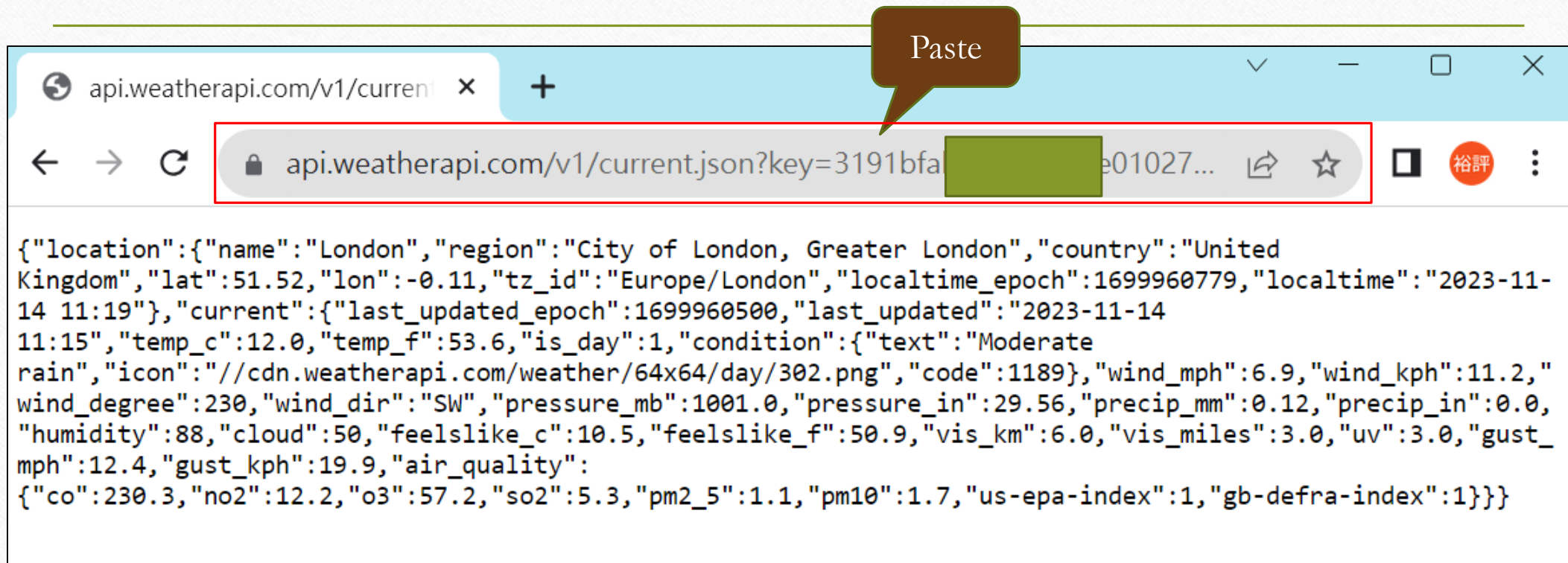
Response Headers

```
{
  "Transfer-Encoding": "chunked",
  "Connection": "keep-alive",
  "Vary": "Accept-Encoding",
  "CDN-PullZone": "93447",
  "CDN-Uid": "8fa3a04a-75d9-4707-8056-b7b33c8ac7fe",
  "CDN-RequestCountryCode": "GB",
  "Age": "0",
  "x-weatherapi-qpm-left": "5000001",
  "CDN-ProxyVer": "1.04",
  "CDN-RequestPullSuccess": "True",
  "CDN-RequestPullCode": "200",
  "CDN-CachedAt": "11/14/2023 11:19:56",
  "CDN-EdgeStorageId": "863",
  "CDN-Status": "200",
  "CDN-RequestId": "54e184433660245ea4695dfe17316888",
  "CDN-Cache": "MISS",
  "Cache-Control": "public, max-age=180",
  "Content-Type": "application/json",
  "Date": "Tue, 14 Nov 2023 11:19:56 GMT",
  "Server": "BunnyCDN-DE1-1079",
  "Via": "1.1 varnish (Varnish/6.0)"
}
```

Response Body

```
{
  "location": {
    "name": "London",
    "region": "City of London, Greater London",
    "country": "United Kingdom",
    "lat": 51.52,
    "lon": -0.11,
    "tz_id": "Europe/London",
    "localtime_epoch": 1699960755,
    "localtime": "2023-11-14 11:19"
  },
  "current": {
    "last_updated_epoch": 1699960500,
    "last_updated": "2023-11-14 11:15",
    "temp_c": 12.0,
    "temp_f": 53.6,
    "is_day": 1,
    "condition": {
      "text": "Moderate rain",
      "icon": "//cdn.weatherapi.com/weather/64x64/day/302.png",
      "code": 1189
    },
    "wind_mph": 6.9,
    "wind_kph": 11.2,
    "wind_degree": 230,
    "wind_dir": "SW",
    "pressure_mb": 1001.0,
    "pressure_in": 29.56,
    "precip_mm": 0.12,
    "precip_in": 0.0,
    "humidity": 88,
    "cloud": 50,
```


Paste the Link into a Browser Address Bar



Select All & Copy



The screenshot shows a web browser window with the address bar displaying `api.weatherapi.com/v1/current.json?key=3191bfabdc3f4c899e01027...`. The main content area displays a JSON object representing weather data for London. The JSON is highlighted in blue, and a brown speech bubble with the word "Copy" is positioned next to it.

```
{
  "location": {
    "name": "London",
    "region": "City of London, Greater London",
    "country": "United Kingdom",
    "lat": 51.52,
    "lon": -0.11,
    "tz_id": "Europe/London",
    "localtime_epoch": 1699960779,
    "localtime": "2023-11-14 11:19"
  },
  "current": {
    "last_updated_epoch": 1699960500,
    "last_updated": "2023-11-14 11:15",
    "temp_c": 12.0,
    "temp_f": 53.6,
    "is_day": 1,
    "condition": {
      "text": "Moderate rain",
      "icon": "http://cdn.weatherapi.com/weather/64x64/day/302.png",
      "code": 1189
    },
    "wind_mph": 6.9,
    "wind_kph": 11.2,
    "wind_degree": 230,
    "wind_dir": "SW",
    "pressure_mb": 1001.0,
    "pressure_in": 29.56,
    "precip_mm": 0.12,
    "precip_in": 0.0,
    "humidity": 88,
    "cloud": 50,
    "feelslike_c": 10.5,
    "feelslike_f": 50.9,
    "vis_km": 6.0,
    "vis_miles": 3.0,
    "uv": 3.0,
    "gust_mph": 12.4,
    "gust_kph": 19.9,
    "air_quality": {
      "co": 230.3,
      "no2": 12.2,
      "o3": 57.2,
      "so2": 5.3,
      "pm2_5": 1.1,
      "pm10": 1.7,
      "us-epa-index": 1,
      "gb-defra-index": 1
    }
  }
}
```


JSON Editor Online

Theme ▾ Features ▾ Help ▾

New document 1



text tree table

```
1 {"location":  
  {"name": "London", "region"  
    London, Greater  
    London", "country": "United  
    Kingdom", "lat": 51.52, "lon": -0.11, "tz_id"  
      ": "Europe/London", "localtime_epoch": 169  
        9960779, "localtime": "2023-11-14  
          11:19"}, "current":  
    {"last_updated_epoch": 1699960500, "last_  
      updated": "2023-11-14  
        11:15", "temp_c": 12.0, "temp_f": 53.6, "is_  
          day": 1, "condition": {"text": "Moderate  
            rain", "icon": "//cdn.weatherapi.com/weat  
              her/64x64/day/302.png", "code": 1189}, "wi  
                nd_mph": 6.9, "wind_kph": 11.2, "wind_degre  
                  e": 230, "wind_dir": "SW", "pressure_mb": 10  
                    01.0, "pressure_in": 29.56, "precip_mm": 0.  
                      12, "precip_in": 0.0, "humidity": 88, "cloud  
                        ": 50, "feelslike_c": 10.5, "feelslike_f": 5
```

Line: 1 Column: 819

Do you want to format the JSON?

Format No thanks

Waiting for thrle.com...

1. Patse

New document 2



text tree table

Copy



Transform



Differences

☐ Compare

>

```
{  
  location : {  
    name : London  
    region : City of London, Greater London  
    country : United Kingdom  
    lat : 51.52  
    lon : -0.11  
    tz_id : Europe/London  
    localtime_epoch : 1699960779  
    localtime : 2023-11-14 11:19  
  }  
  current : {  
    last_updated_epoch : 1699960500  
    last_updated : 2023-11-14 11:15  
    temp_c : 12.0  
    temp_f : 53.6  
    is_day : 1  
    condition : {  
      text : Moderate rain
```


JSON Editor Online

Theme ▾ Features ▾ Help ▾

New document 1



text tree table

```
1 {"location":
  {"name": "London", "region": "City of
  London, Greater
  London", "country": "United
  Kingdom", "lat": 51.52, "lon": -0.11, "tz_id
  ": "Europe/London", "localtime_epoch": 169
  9960779, "localtime": "2023-11-14
  11:19"}, "current":
  {"last_updated_epoch": 1699960500, "last_
  updated": "2023-11-14
  11:15", "temp_c": 12.0, "temp_f": 53.6, "is_
  day": 1, "condition": {"text": "Moderate
  rain", "icon": "//cdn.weatherapi.com/weat
  her/64x64/day/302.png", "code": 1189}, "wi
  nd_mph": 6.9, "wind_kph": 11.2, "wind_degre
  e": 230, "wind_dir": "SW", "pressure_mb": 10
  01.0, "pressure_in": 29.56, "precip_mm": 0.
  12, "precip_in": 0.0, "humidity": 88, "cloud
  ": 50, "feelslike_c": 10.5, "feelslike_f": 5
```

Line: 1 Column: 819

Do you want to format the JSON?

Format

No thanks

Copy



Transform



Differences

☐ Compare

New document 2



text tree table

```
>
  cloud : 50
  feelslike_c : 10.5
  feelslike_f : 50.9
  vis_km : 6.0
  vis_miles : 3.0
  uv : 3.0
  gust_mph : 12.4
  gust_kph : 19.9
  air_quality : {
    co : 230.3
    no2 : 12.2
    o3 : 57.2
    so2 : 5.3
    pm2_5 : 1.1
    pm10 : 1.7
    us-epa-index : 1
    gb-defra-index : 1
  }
}
```

Exercise 9-1

- Get the weather information from weather API and show them on the debug window in Node-RED.

http request



Edit http request node

Delete Cancel Done

Properties

Method GET

URL https://api.weatherapi.com/v1/current.json?key=31

Payload Ignore

☐ Enable secure (SSL/TLS) connection

☐ Use authentication

☐ Enable connection keep-alive

☐ Use proxy

☐ Only send non-2xx responses to Catch n

☐ Disable strict HTTP parsing

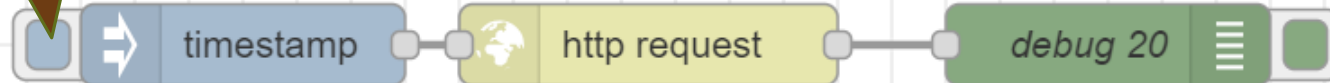
Return a parsed JSON object

Your weather API

A parsed JSON object

Triger

Click

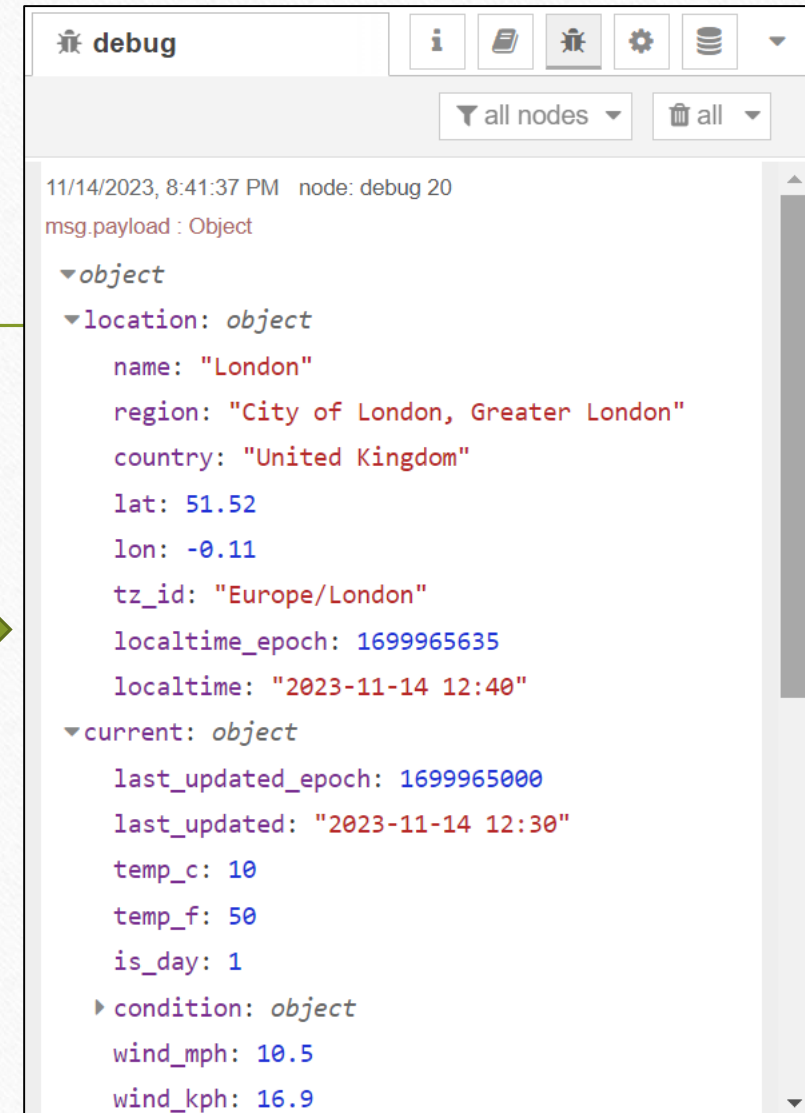
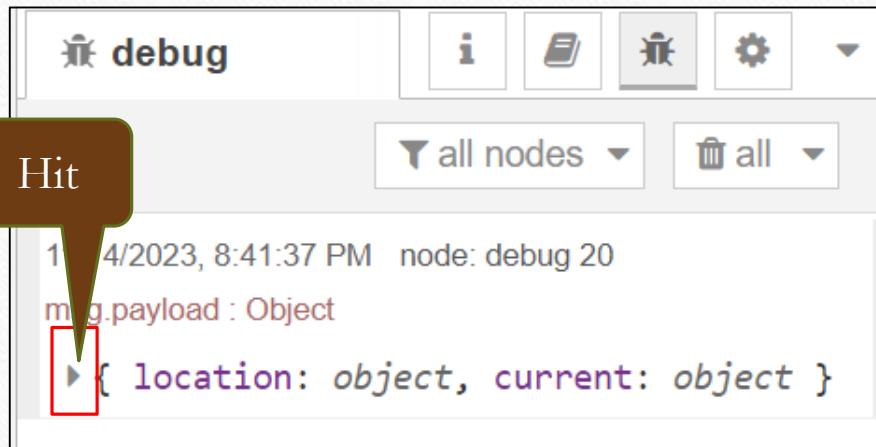


debug

all nodes all

11/14/2023, 8:41:37 PM node: debug 20
msg.payload : Object
▶ { location: object, current: object }

Expand



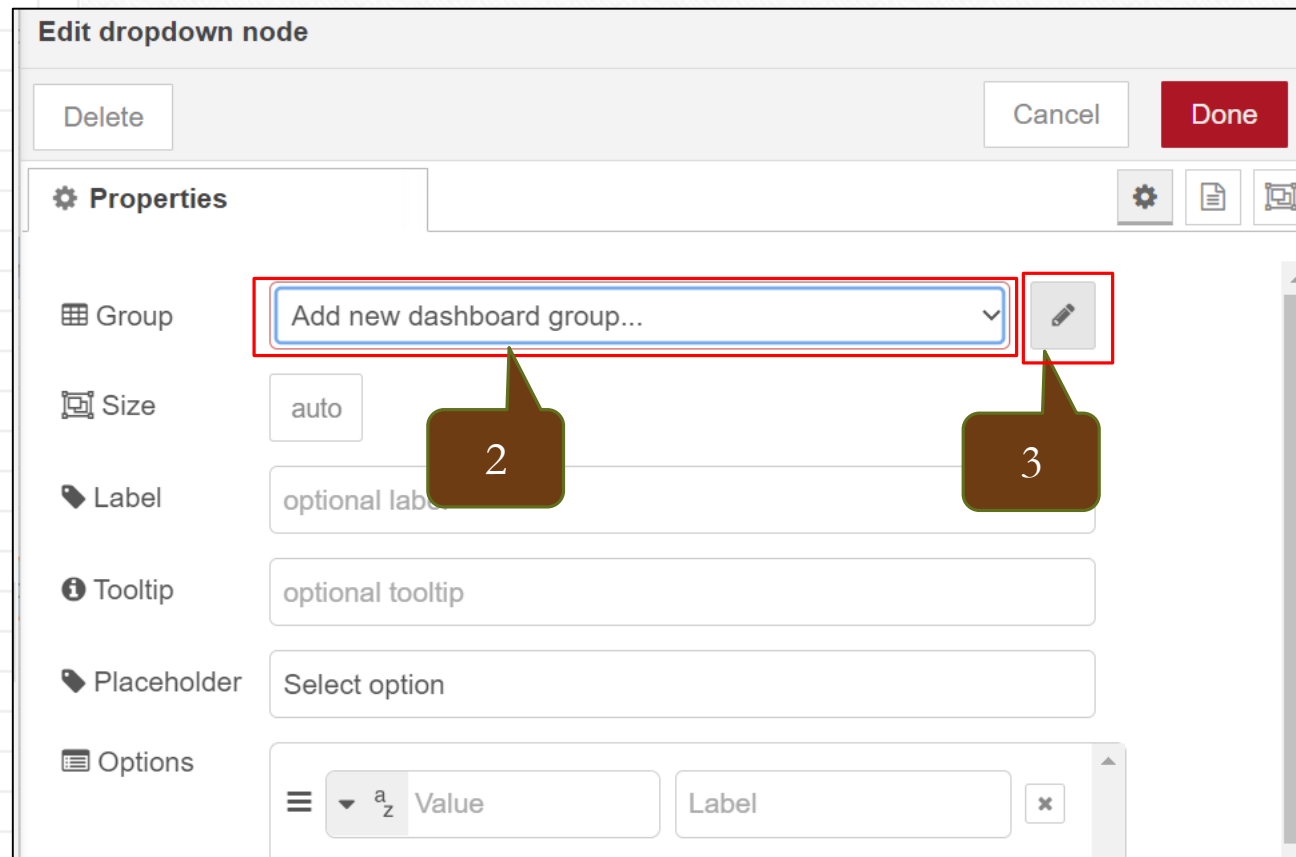
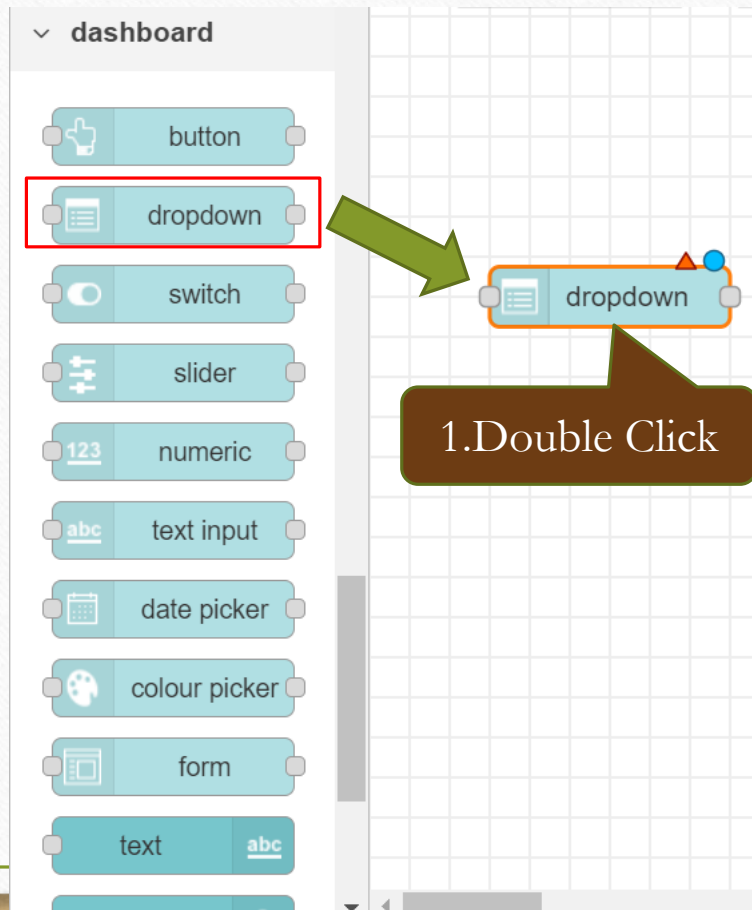
Homework 1

- Test the weather API for seven different cities.
- Taipei
- London
- Singapur
- Paris
- Taichung
- Taoyuan
- New York

Exercise 9-2

- Build a weather station.

Step 1




Edit dropdown node > **Add new dashboard group config node**

Cancel Add

Properties

Name Default

Tab Add new dashboard tab... 

Class Optional CSS class name(s) for widget

Width 6

1 2

Edit dropdown node > Add new dashboard group config node > **Add new dashboard tab config node**

Cancel Add

Properties

Name Home 9-2

Icon dashboard

State ☒ Enabled

Nav. Menu

The Icon field is for a Font Awesome icon (e.g., 'fa-sunny'). You can use the icon picker to select an icon.


3 4

Edit dropdown node > **Add new dashboard group config node**

Cancel Add

Properties

Name Default

Tab Home 9-2 

Class Optional CSS class name(s) for widget

Width 6

5

Flow 6

Edit dropdown node > Add new dashboard group config node

Cancel

Add

Properties

Name

Default

Tab

Home 9-2

</> Class

Optional CSS class name(s) for widget

↔ Width

6

☒ Display group name

☐ Allow group to be collapsed

Add

timestamp

http req

dropdown

Edit dropdown node

Delete

Cancel

Done

Properties

Tooltip

optional tooltip

Placeholder

Select option

Options

1

▼ a_z Taipei

Taipei

×

3

▼ a_z London

London

×

5

▼ a_z Singapur

Singapur

×

7

▼ a_z Paris

Paris

×

9

▼ a_z Taichung

Taichung

×

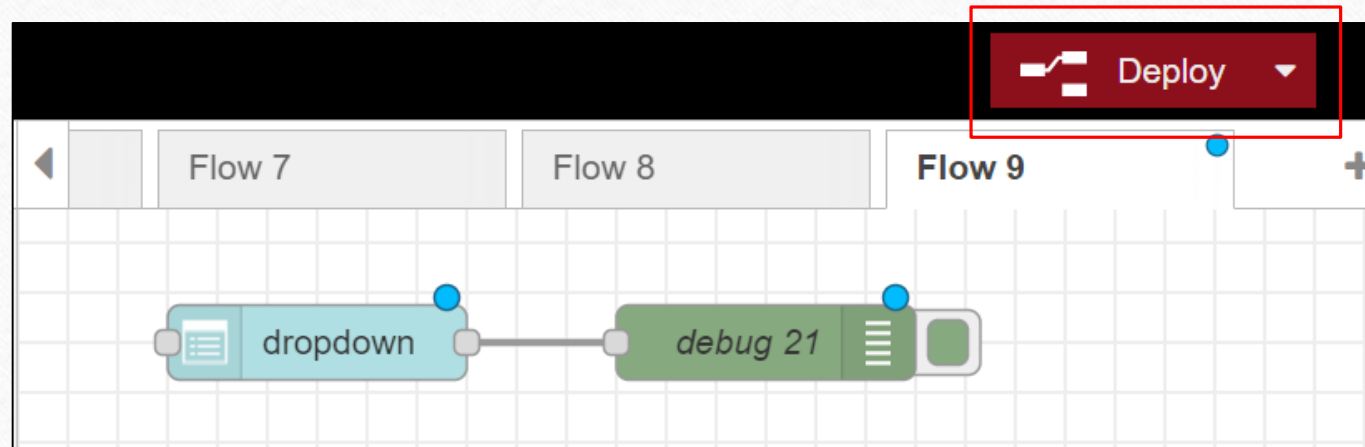
+ option

2,4,6,8

multiple selections from list: ☐

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

Add a debug node and Deploy

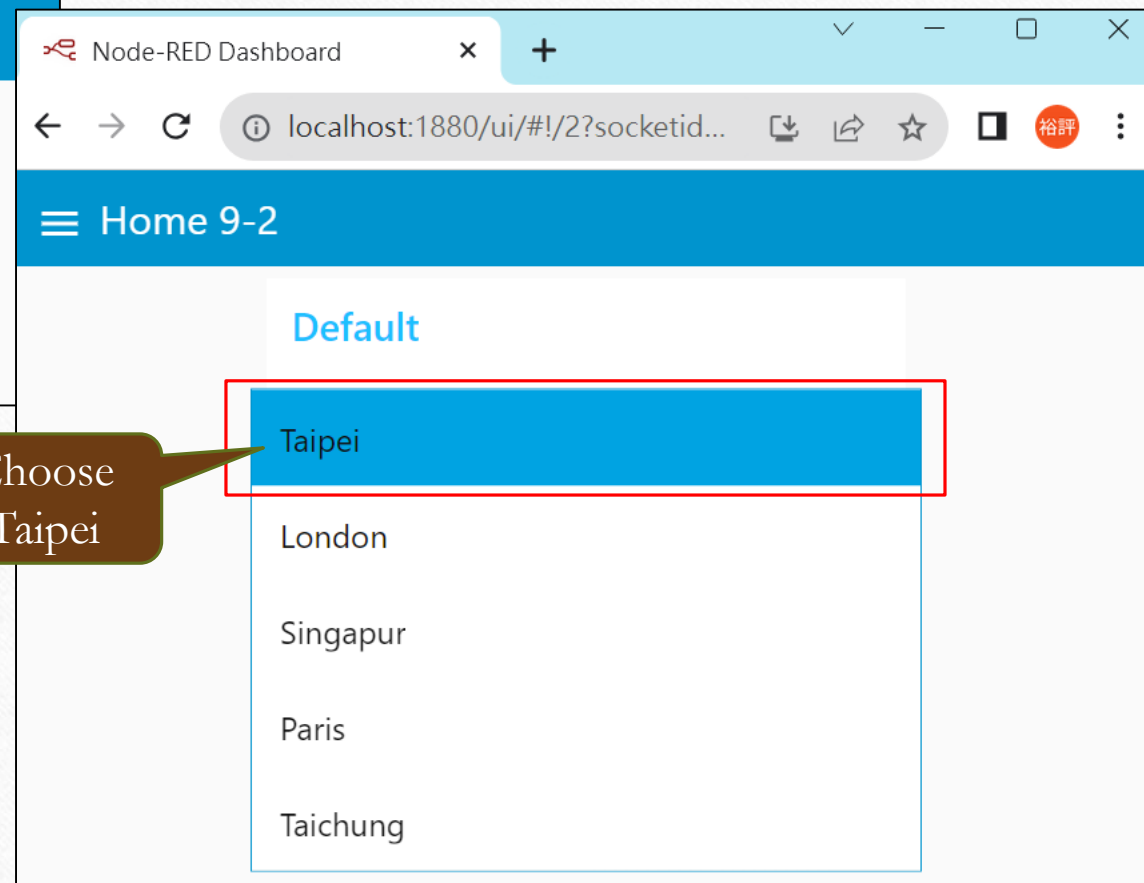
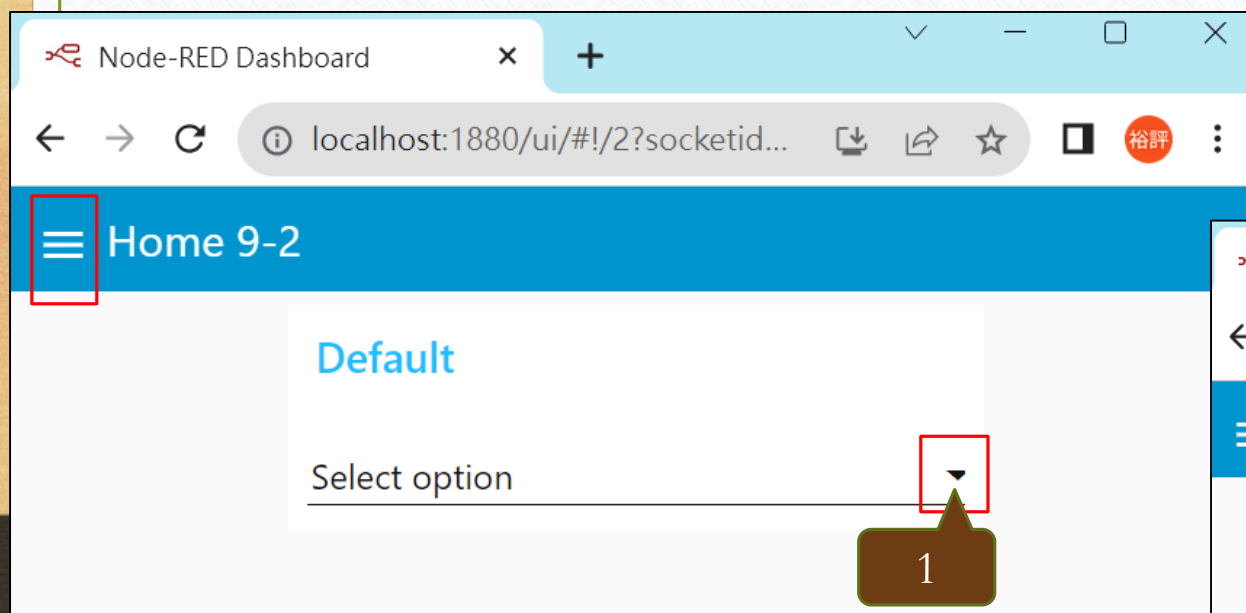


Go to Dashboard

The image consists of two side-by-side screenshots of a web application interface, illustrating the steps to navigate to the dashboard. The interface features a top navigation bar with icons for information, help, debug messages, configuration nodes, context data, and a dashboard icon. The left screenshot shows the 'debug' tab selected, with a dropdown menu open. The 'Dashboard' option, represented by a bar chart icon, is highlighted at the bottom of the menu. The right screenshot shows the 'dashboard' tab selected, with a 'Layout' button and a 'Theme' button visible. Both buttons are highlighted, indicating they are part of the navigation process.

Left Screenshot: The 'debug' tab is active. A dropdown menu is open, showing options: Information, Help, Debug messages, Configuration nodes, Context Data, and **Dashboard** (highlighted with a red box and labeled 2). A red box around the dropdown arrow is labeled 1.

Right Screenshot: The 'dashboard' tab is active. The 'Layout' button and 'Theme' button are visible. Both buttons are highlighted with red boxes and labeled 3.



Home 9-2

Default

Taipei

debug

all nodes

all

11/15/2023, 1:01:08 AM node: debug 21

msg.payload : string[6]

"Taipei"

Node-RED Dashboard

localhost:1880/ui/#!/2?socketid...

Home 9-2

Default

London

Choose London

11/15/2023, 1:03:45 AM node: debug 21

msg.payload : string[6]

"London"

Node-RED Dashboard

localhost:1880/ui/#!/2?socketid...

Home 9-2

Default

Taichung

all nodes

all

11/15/2023, 1:01:08 AM node: debug 21
msg.payload : string[6]
"Taipei"

11/15/2023, 1:03:45 AM node: debug 21
msg.payload : string[6]
"London"

11/15/2023, 1:05:45 AM node: debug 21
msg.payload : string[8]
"Singapur"

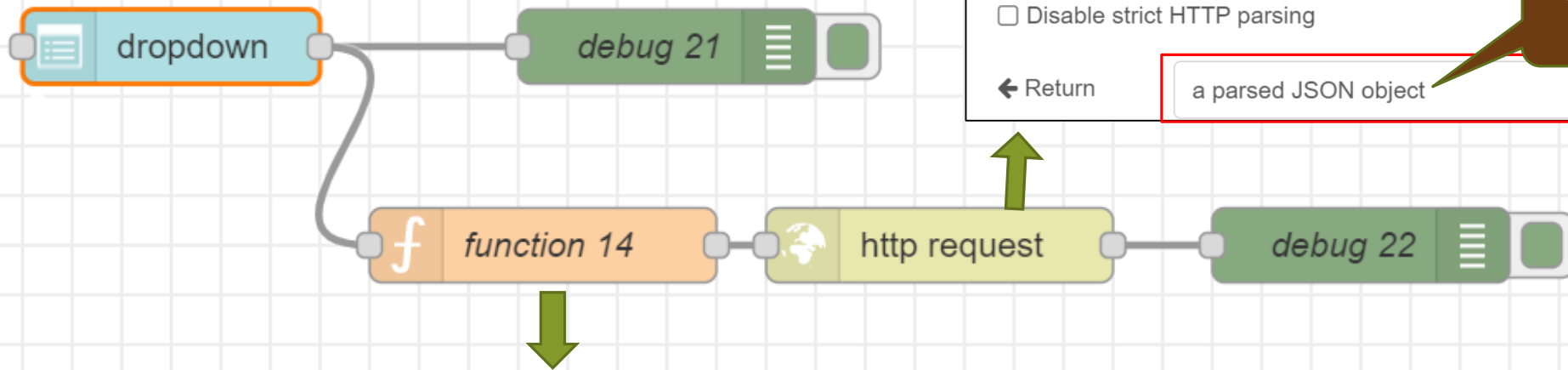
11/15/2023, 1:05:49 AM node: debug 21
msg.payload : string[5]
"Paris"

11/15/2023, 1:05:52 AM node: debug 21
msg.payload : string[8]
"Taichung"

Choose
Taichung



Add nodes



Method GET

URL http://

Payload Ignore

☐ Enable secure (SSL/TLS) connection

☐ Use authentication

☐ Enable connection keep-alive

☐ Use proxy

☐ Only send non-2xx responses to Catch node

☐ Disable strict HTTP parsing

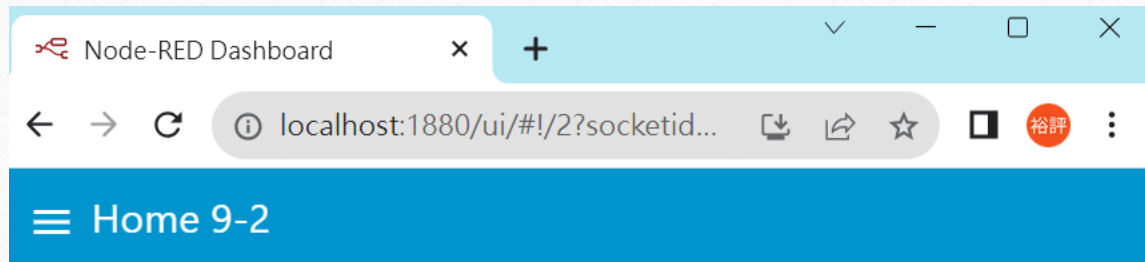
Return a parsed JSON object

blank

a parsed JSON object

```
var city=msg.payload;
msg.url="https://api.weatherapi.com/v1/current.json?key=3xxx11&q="+city+"&aqi=yes";
return msg;
```

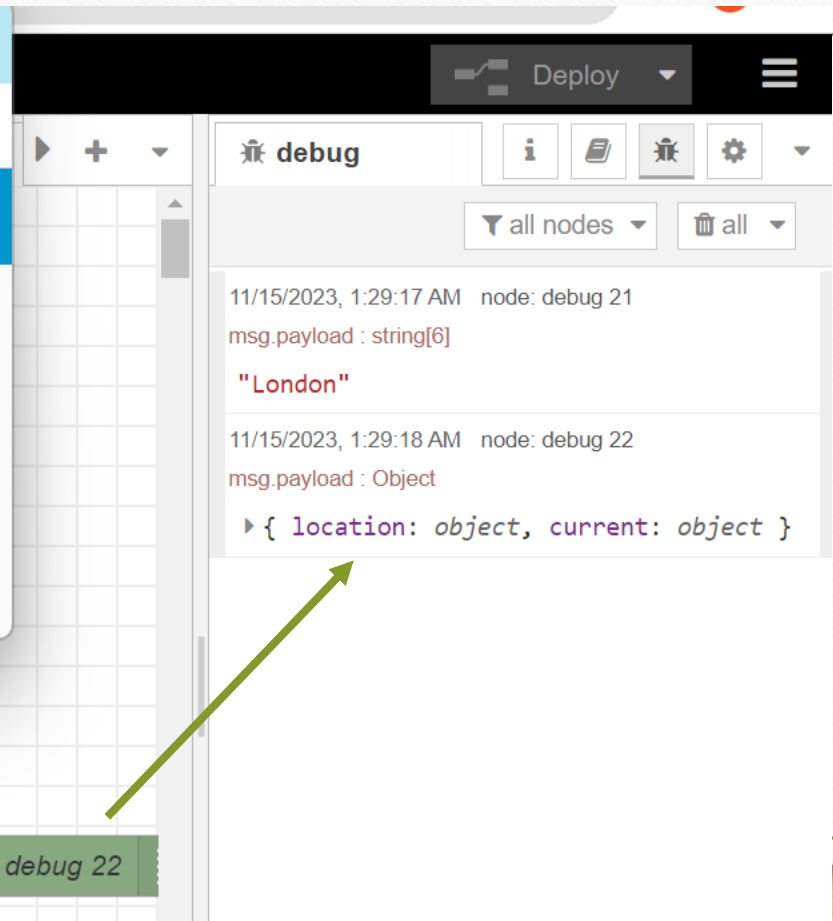
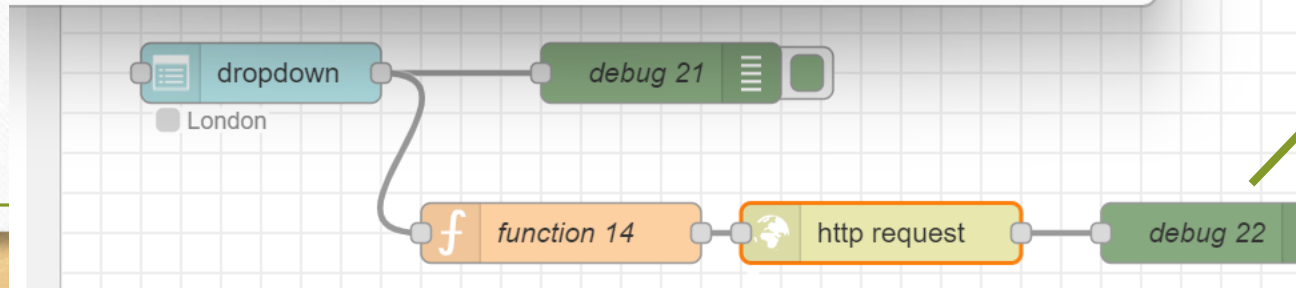
Test with choosing London



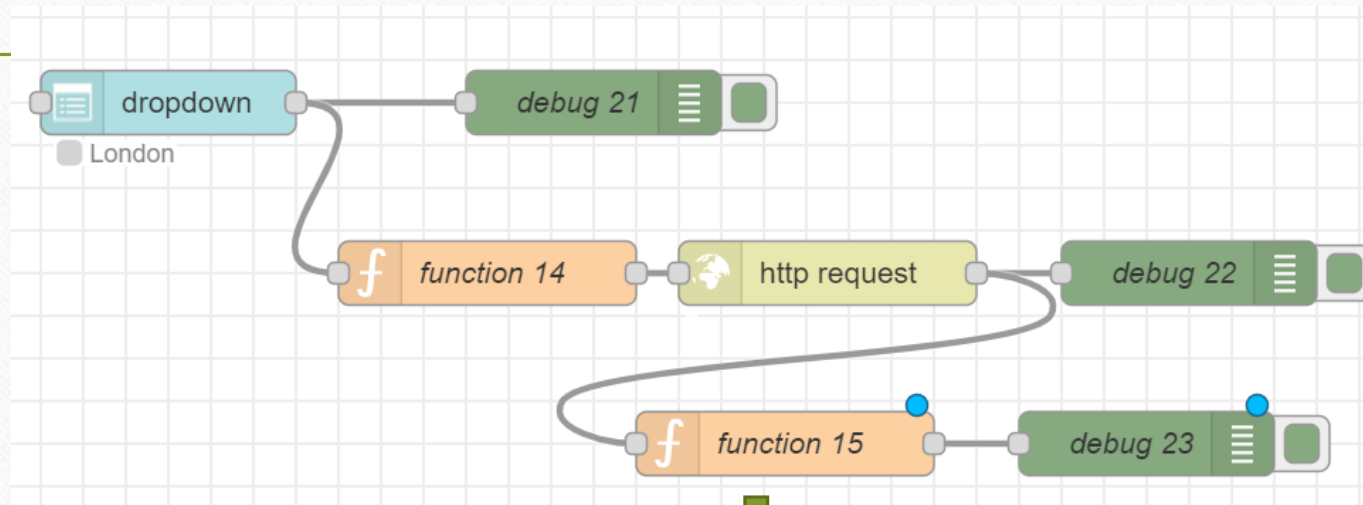
Default

London

Choose
London



Add a function node and a debug node



```
var obj = msg.payload;  
var current = obj.current;  
msg.payload = current;  
return msg;
```

Choose Taipei

The screenshot displays the Node-RED web interface in a browser window. The browser's address bar shows the URL `localhost:1880/ui/#!/2?socketid...`. The interface includes a top navigation bar with a hamburger menu and the text "Home 9-2". Below this, a "Default" tab is active. A red box highlights a dropdown menu containing the text "Taipei". A callout bubble points to this dropdown with the text "Choose Taipei".

The main workspace shows a flow diagram with the following components:

- A "dropdown" node with a "Taipei" option selected.
- A "function 14" node connected to the dropdown.
- An "http request" node connected to "function 14".
- A "debug 21" node connected to the "dropdown" node.
- A "function 15" node connected to the "http request" node.
- A "debug 22" node connected to the "http request" node.
- A "debug 23" node connected to "function 15".

On the right side, the "debug" console is open, showing a list of messages. A red box highlights the "object" property of the third message, and a callout bubble points to it with the text "Expand". The expanded object shows the following data:

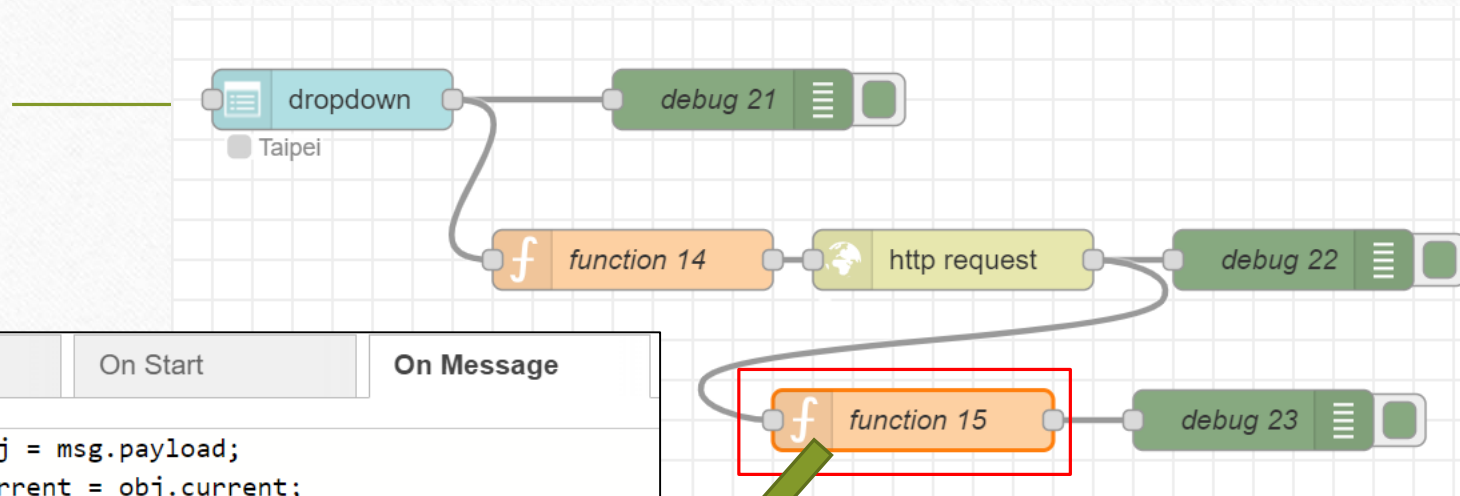
```
last_updated_epoch: 1699983000
last_updated: "2023-11-15 01:30"
temp_c: 19
temp_f: 66.2
is_day: 0
condition: object
wind_mph: 11.9
wind_kph: 19.1
```

▼ *object*

last_updated_epoch:
1699983000
last_updated: "2023-11-15
01:30"
temp_c: 19
temp_f: 66.2
is_day: 0
▶ condition: *object*
wind_mph: 11.9
wind_kph: 19.1
wind_degree: 100
wind_dir: "E"
pressure_mb: 1023
pressure_in: 30.21
precip_mm: 0.15
precip_in: 0.01
humidity: 88
cloud: 75
feelslike_c: 19
feelslike_f: 66.2

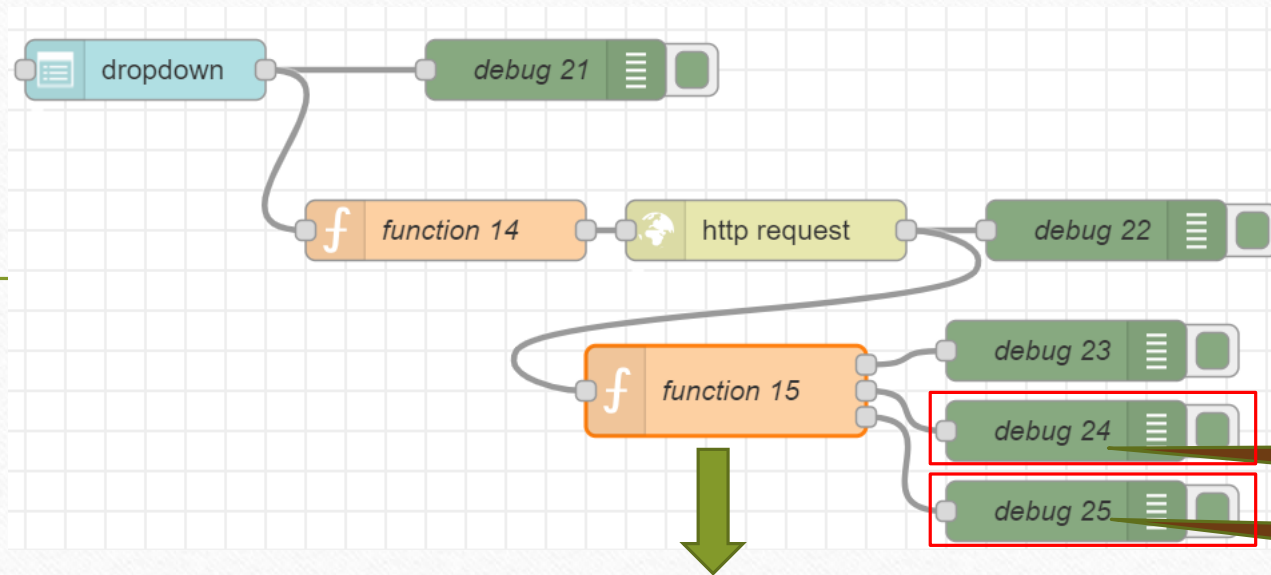
precip_in: 0.01
humidity: 88
cloud: 75
feelslike_c: 19
feelslike_f: 66.2
vis_km: 10
vis_miles: 6
uv: 1
gust_mph: 13.2
gust_kph: 21.3
▼ air_quality: *object*
co: 270.4
no2: 5.7
o3: 88
so2: 7
pm2_5: 2.1
pm10: 3.6
us-epa-index: 1
gb-defra-index: 1

Edit function



Setup On Start On Message

```
1 var obj = msg.payload;
2 var current = obj.current;
3 var temp_c = current.temp_c;
4 var humidity = current.humidity;
5 var uv = current.uv;
6 msg={};
7 let msg1={};
8 let msg2={};
9 msg.topic = "temperature";
10 msg.payload = temp_c;
11 msg1.topic = "humidity";
12 msg1.payload = humidity;
13 msg2.topic = "uv";
14 msg2.payload = uv;
15 return [msg, msg1, msg2];
```



Deploy

Add

Add

Properties

Name: function 15

Setup | On Start | On Message | On Stop

Outputs: 3 | Timeout: 0

3

Import as

Node-RED Dashboard x +

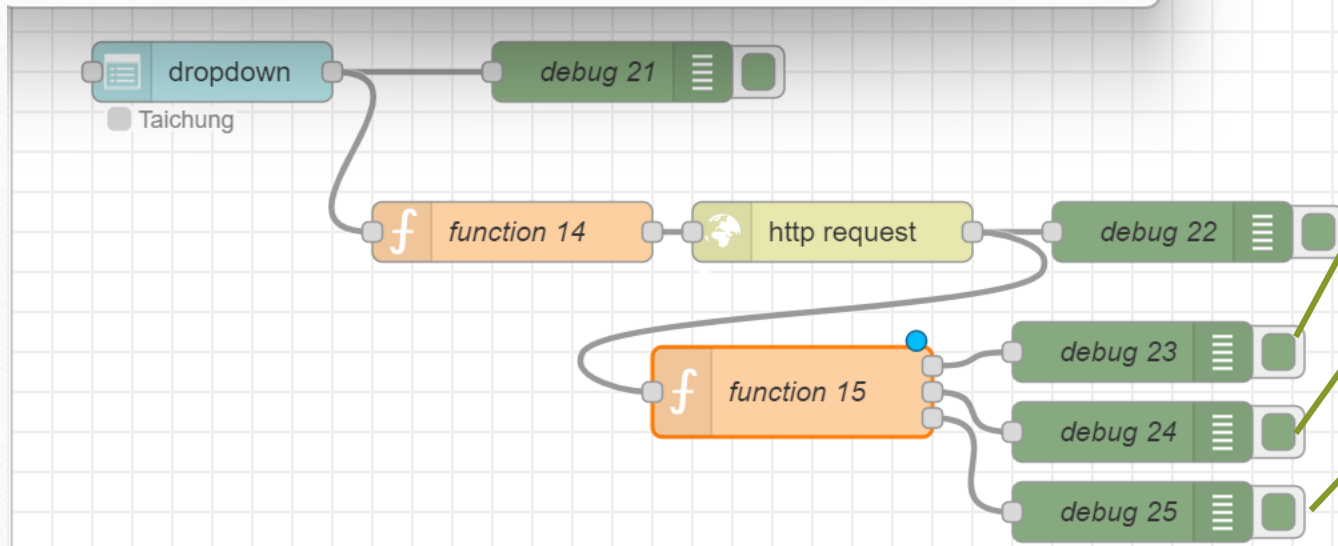
localhost:1880/ui/#!/?socketid...

Home 9-2

Default

Taichung

Choose
Taichung



Deploy

debug

all nodes all

11/15/2023, 2:24:07 AM node: debug 21
msg.payload : string[8]
"Taichung"

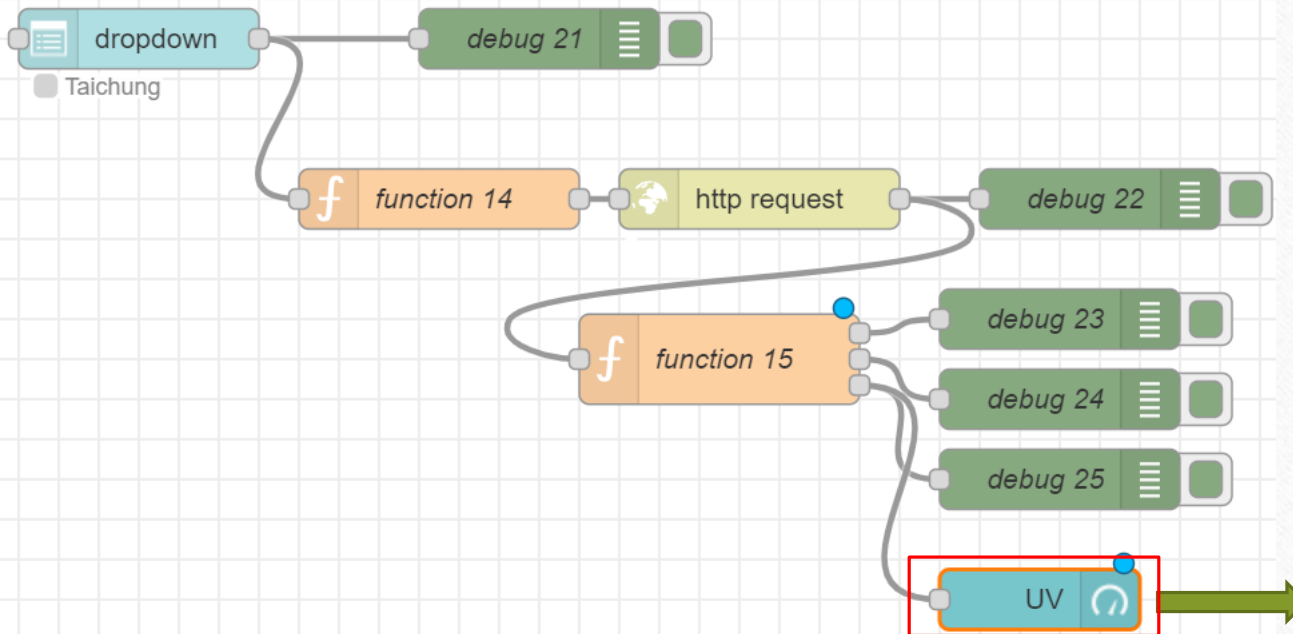
11/15/2023, 2:24:08 AM node: debug 22
msg.payload : Object
{ location: object, current: object }

11/15/2023, 2:24:08 AM node: debug 23
temperature : msg.payload : number
18

11/15/2023, 2:24:08 AM node: debug 24
humidity : msg.payload : number
94

11/15/2023, 2:24:08 AM node: debug 25
uv : msg.payload : number
1

Add a gauge



Edit gauge node

Delete Cancel **Done**

Properties

Group [Home 9-2] Default

Size auto

Type Gauge

Label UV

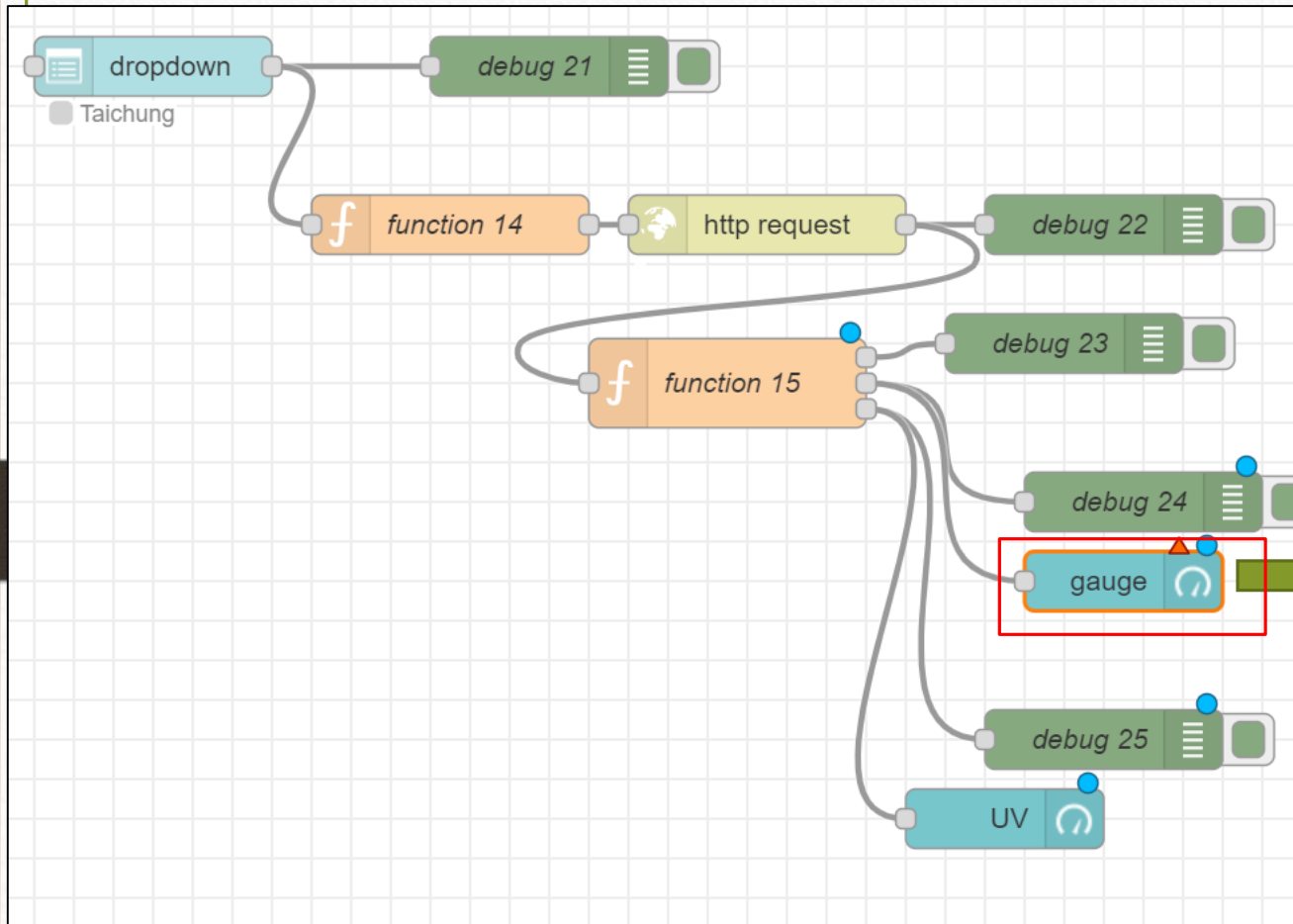
Value format {{value}}

Units units

Range min 0 max 10

Home 9-2

UV



Edit gauge node

DeleteCancelDone

Properties

Group

[Home 9-2] Default

Size

auto

Type

Gauge

Label

Humidity

Value format

{{value}}

Units

units

Range

min 0

max 100

Colour gradient

Sectors

0

...

optional

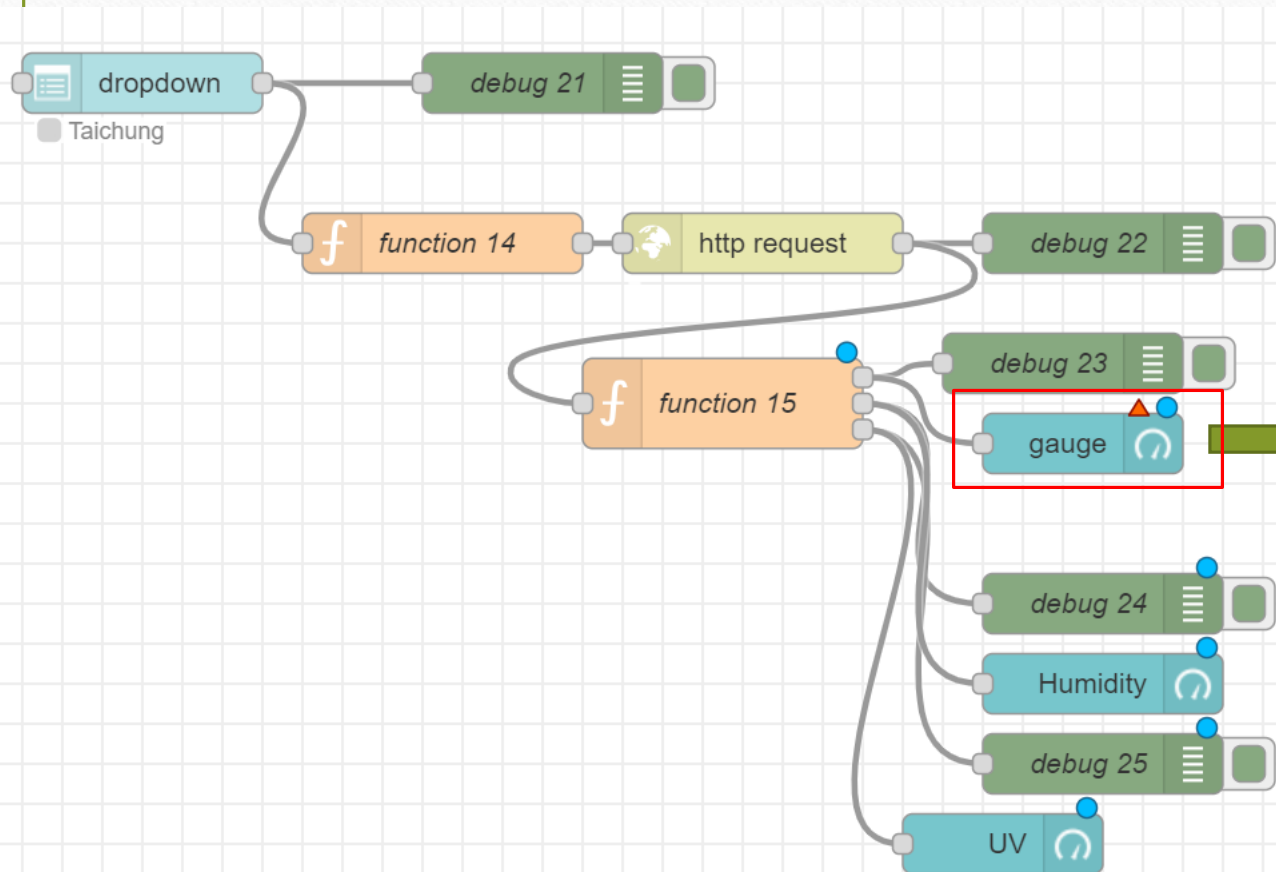
...

optional

...

100

Enabled



Edit gauge node

DeleteCancelDone

Properties

Group

[Home 9-2] Default

Size

auto

Type

Gauge

Label

Temperature

Value format

{{value}}

Units

units


Range

min -20 max 50

Colour gradient

Sectors


-20 ... optional ... optional ... 50

 dashboard



Layout

Site


Theme 

Tabs & Links




+ tab


+ link

▼  Homework 5-1

>  Default

▼  Home 7-3


>  Default

▼  Home 9-2

▼  Default


 dropdown

 Temperature

 Humidity

 UV

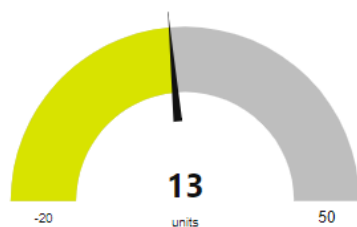
Move to the top

 Deploy ▼

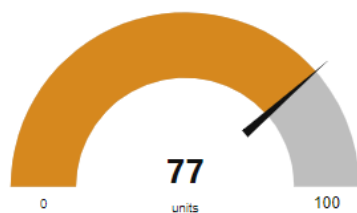
Default

Paris

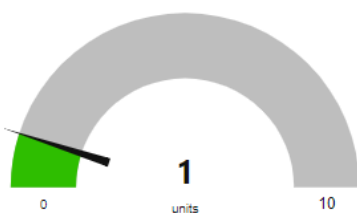
Temperature



Humidity



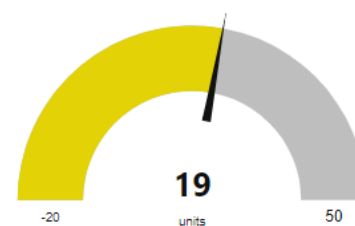
UV



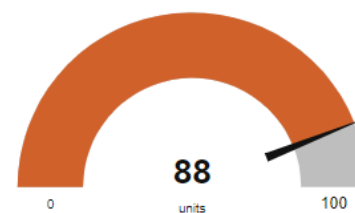
Default

Taipei

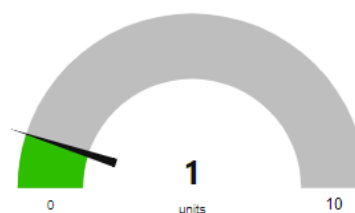
Temperature



Humidity



UV



Homework 2: Design ten cities weather station

- London
- Singapur
- Shanghai
- Taipei
- Taichung
- Taoyuan
- New York
- ??
- ??
- ??