物聯網實務

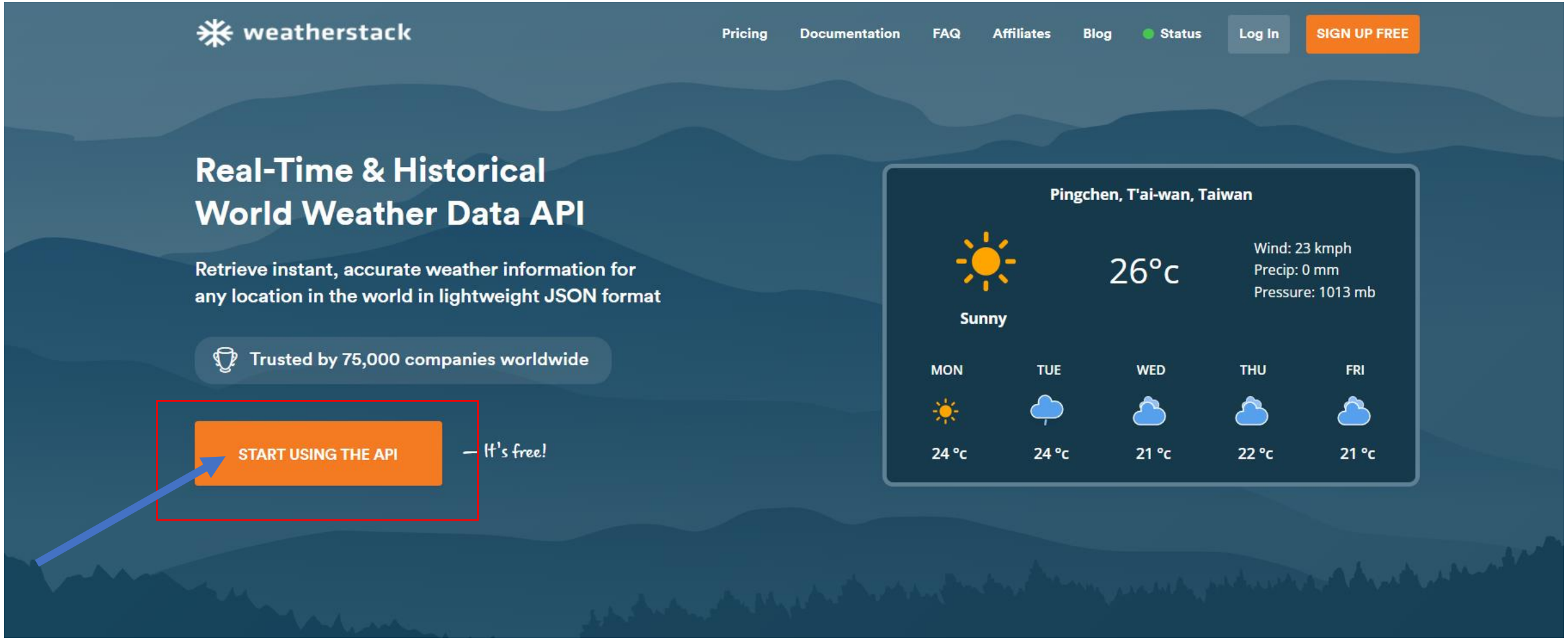
11_23

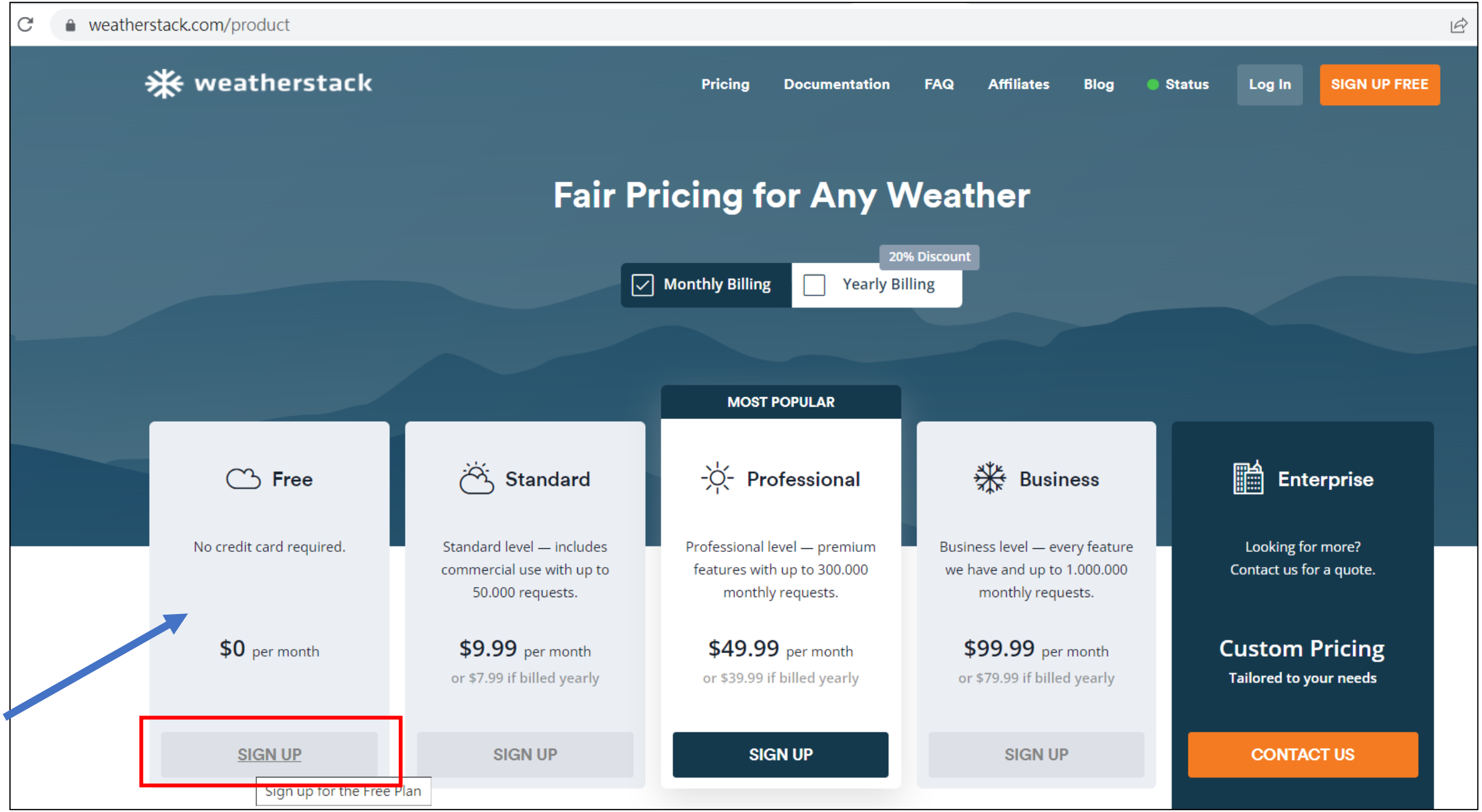
廖裕評

Building a weather station

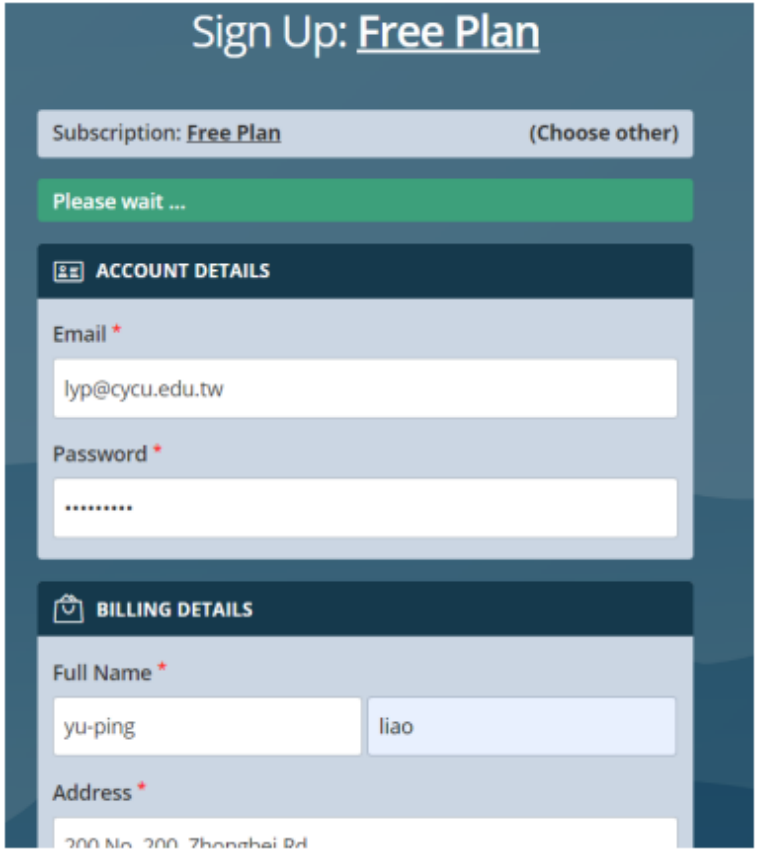
- Weather APIs give detailed information beyond just temperature, precipitation and humidity, but also UV index and pollution levels. Time series data lets you discover trends and patterns, and plan events based on weather conditions. APIs can also be a useful resource for disaster management as some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

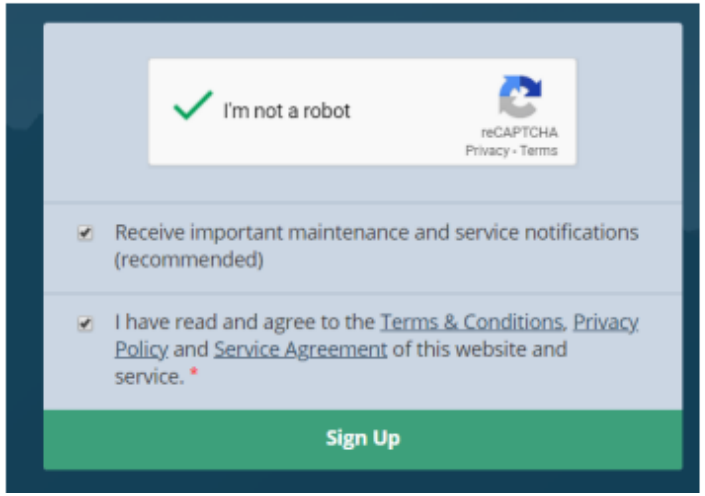
https://weatherstack.com/



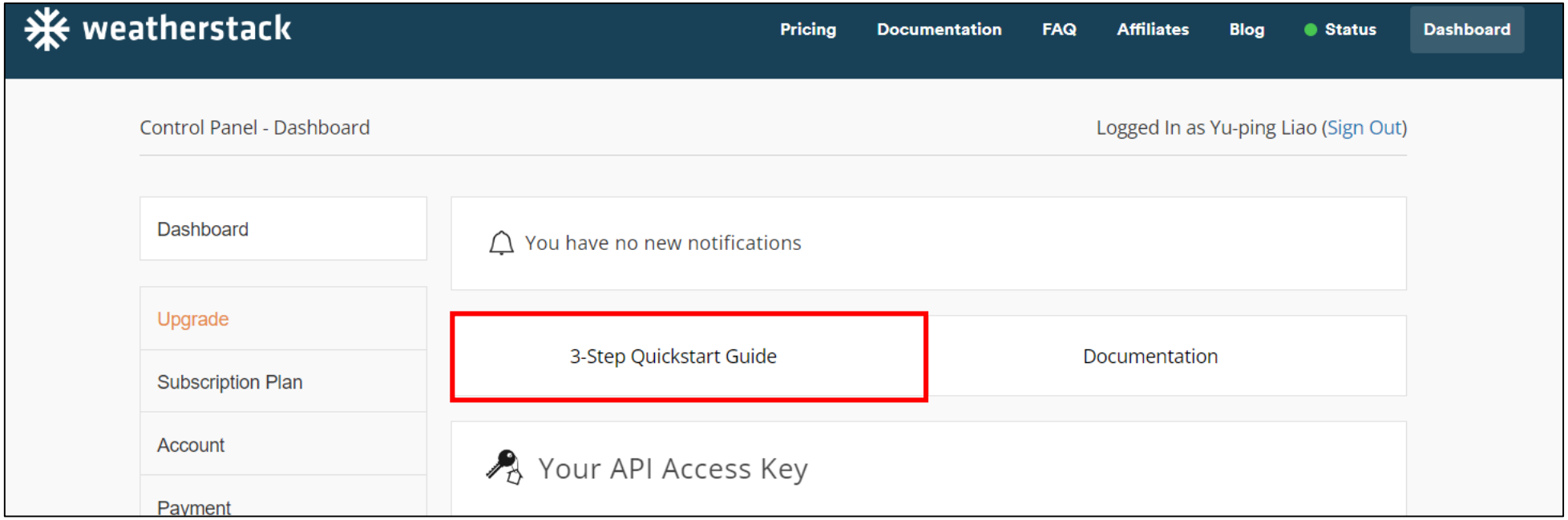

weatherstack.com/product
weatherstack
Pricing
Documentation
FAQ
Affiliates
Blog
Status
Log In
SIGN UP FREE
Fair Pricing for Any Weather
20% Discount
Monthly Billing
Yearly Billing
MOST POPULAR
Free
No credit card required.
$0 per month
SIGN UP
Standard
Standard level — includes commercial use with up to 50.000 requests.
$9.99 per month
or $7.99 if billed yearly
SIGN UP
Professional
Professional level — premium features with up to 300.000 monthly requests.
$49.99 per month
or $39.99 if billed yearly
SIGN UP
Business
Business level — every feature we have and up to 1.000.000 monthly requests.
$99.99 per month
or $79.99 if billed yearly
SIGN UP
Enterprise
Looking for more?
Contact us for a quote.
Custom Pricing
Tailored to your needs
CONTACT US
Sign up for the Free Plan

Sign Up:Free Plan

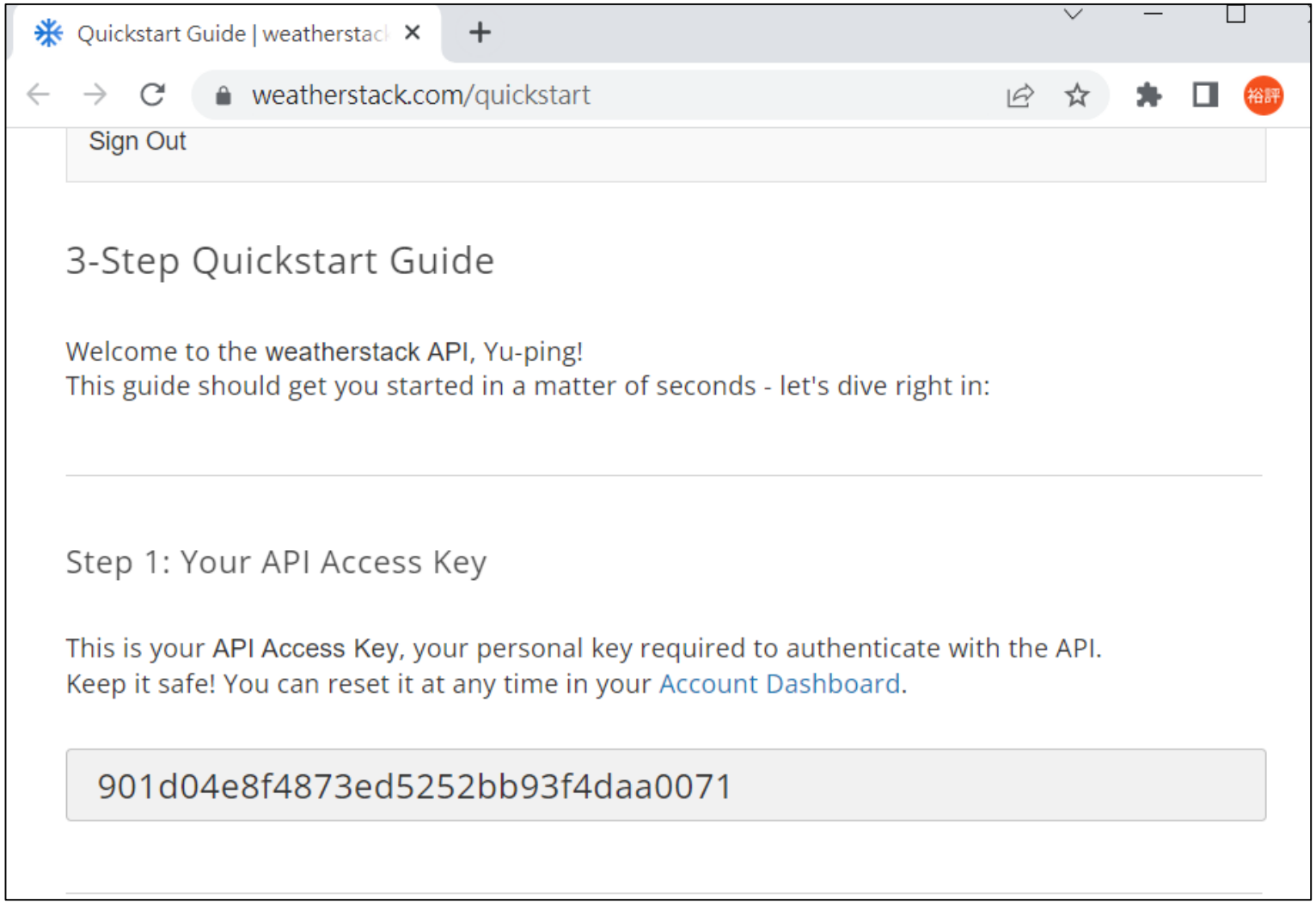




3-Step Quickstart Guide



Step 1 : Your API Access Key



Step 2 : API Endpoints

Step 2: API Endpoints

There are several API endpoints to choose from:

- **Current Weather:** Get current weather data.
- **Historical Weather:** Get historical weather data.
- **Historical Time-Series:** Get historical time-series weather data.
- **Weather Forecast:** Get weather forecast for up to 14 days.
- **Location Lookup:** Look up one or multiple locations.

Base URL: API requests start out with the following base URL:

```
http://api.weatherstack.com/
```

Make API Request: Let's try making a few simple API requests for the current, historical and forecast weather data endpoints. Take a look at the box below and click the API requests to open them in your browser.

Current Weather API

Make API Request

required and **optional**

CURRENT | HISTORICAL | TIME-SERIES | FORECAST

```
// Current Weather API Endpoint

http://api.weatherstack.com/current
    ? access_key = YOUR_ACCESS_KEY
    & query = New York

// optional parameters:

    & units = m
    & language = en
    & callback = MY_CALLBACK
```

HISTORICAL

Make API Request

required and optional

CURRENT | HISTORICAL | TIME-SERIES | FORECAST

```
// Historical Weather API Endpoint

http://api.weatherstack.com/historical
    ? access_key = YOUR_ACCESS_KEY
    & query = New York
    & historical_date = 2015-21-01

// optional parameters:

    & hourly = 1
    & interval = 3
    & units = m
    & language = en
    & callback = MY_CALLBACK
```

TIME-SERIES

Make API Request

required and optional

CURRENT | HISTORICAL | TIME-SERIES | FORECAST

```
// Historical Weather Time-Series:

http://api.weatherstack.com/historical
    ? access_key = YOUR_ACCESS_KEY
    & query = New York
    & historical_date_start = 2015-10-21
    & historical_date_end = 2015-10-25

// optional parameters:

    & hourly = 1
    & interval = 3
    & units = m
    & language = en
    & callback = MY_CALLBACK
```

FORECAST

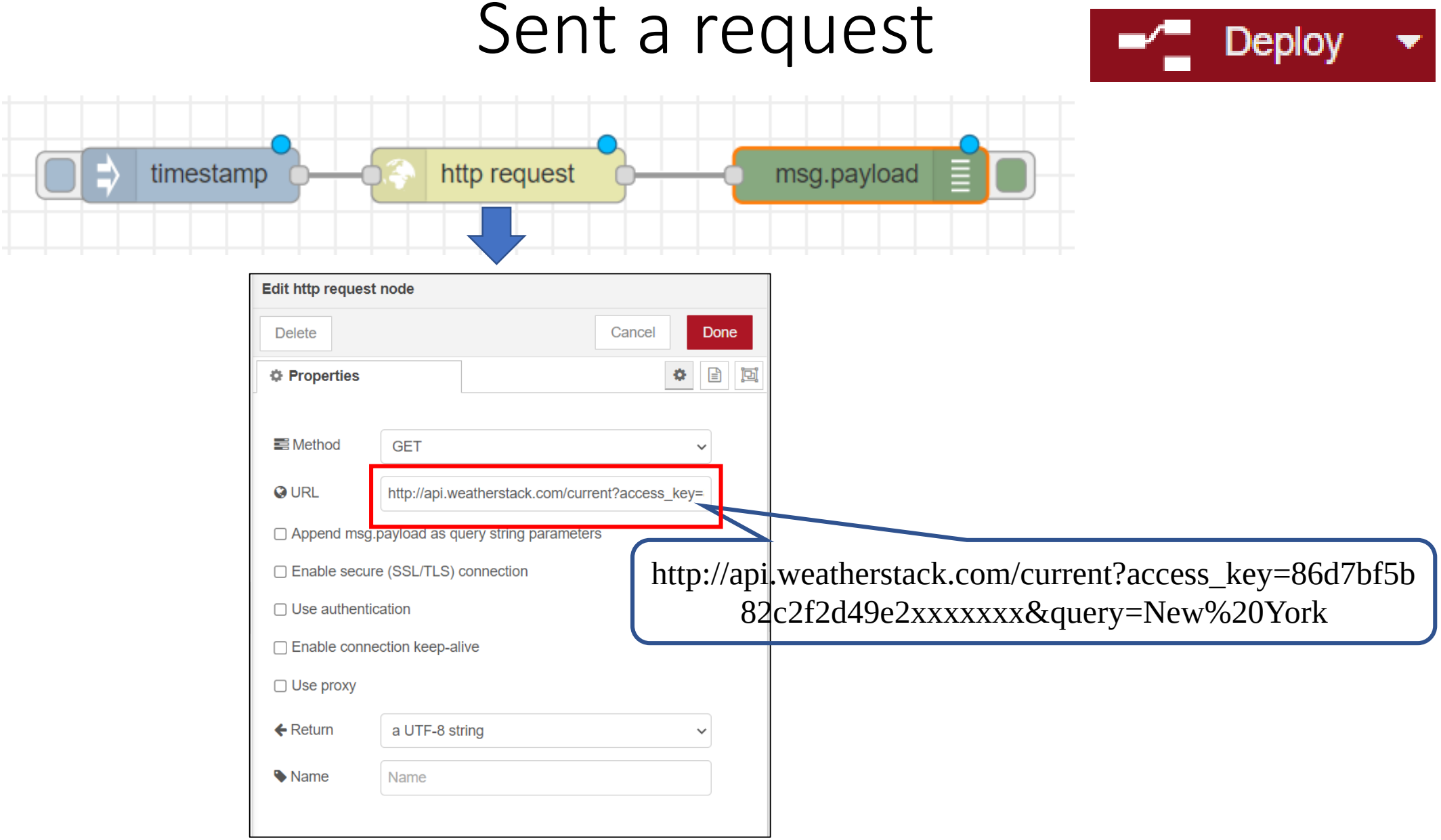
Make API Request

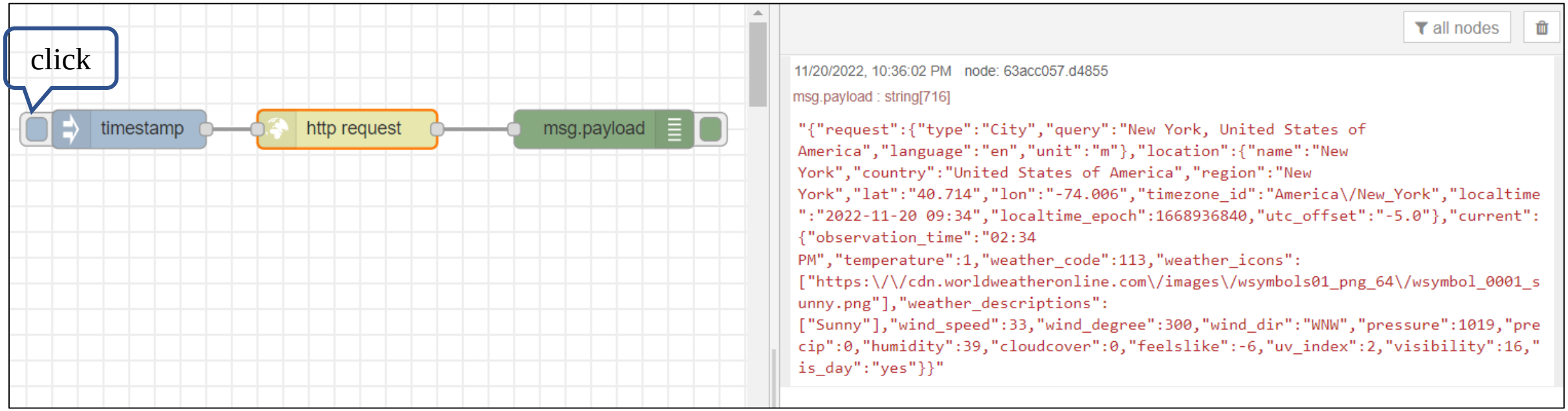
required and **optional**

CURRENT | HISTORICAL | TIME-SERIES | FORECAST

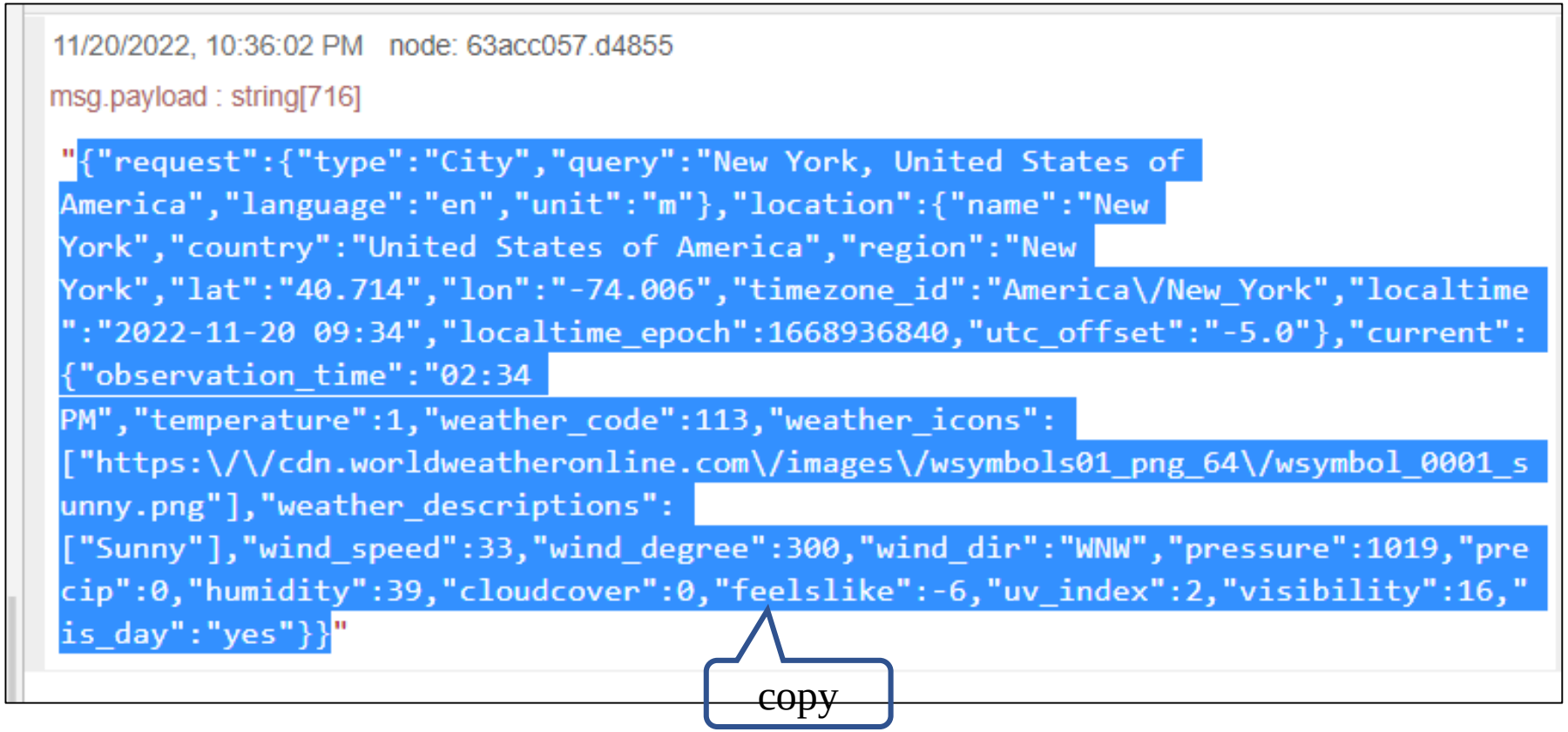
```
// Weather Forecast API Endpoint

http://api.weatherstack.com/forecast
    ? access_key = YOUR_ACCESS_KEY
    & query = New York

// optional parameters:

    & forecast_days = 7
    & hourly = 1
    & interval = 3
    & units = m
    & language = en
    & callback = MY_CALLBACK
```

Sent a request

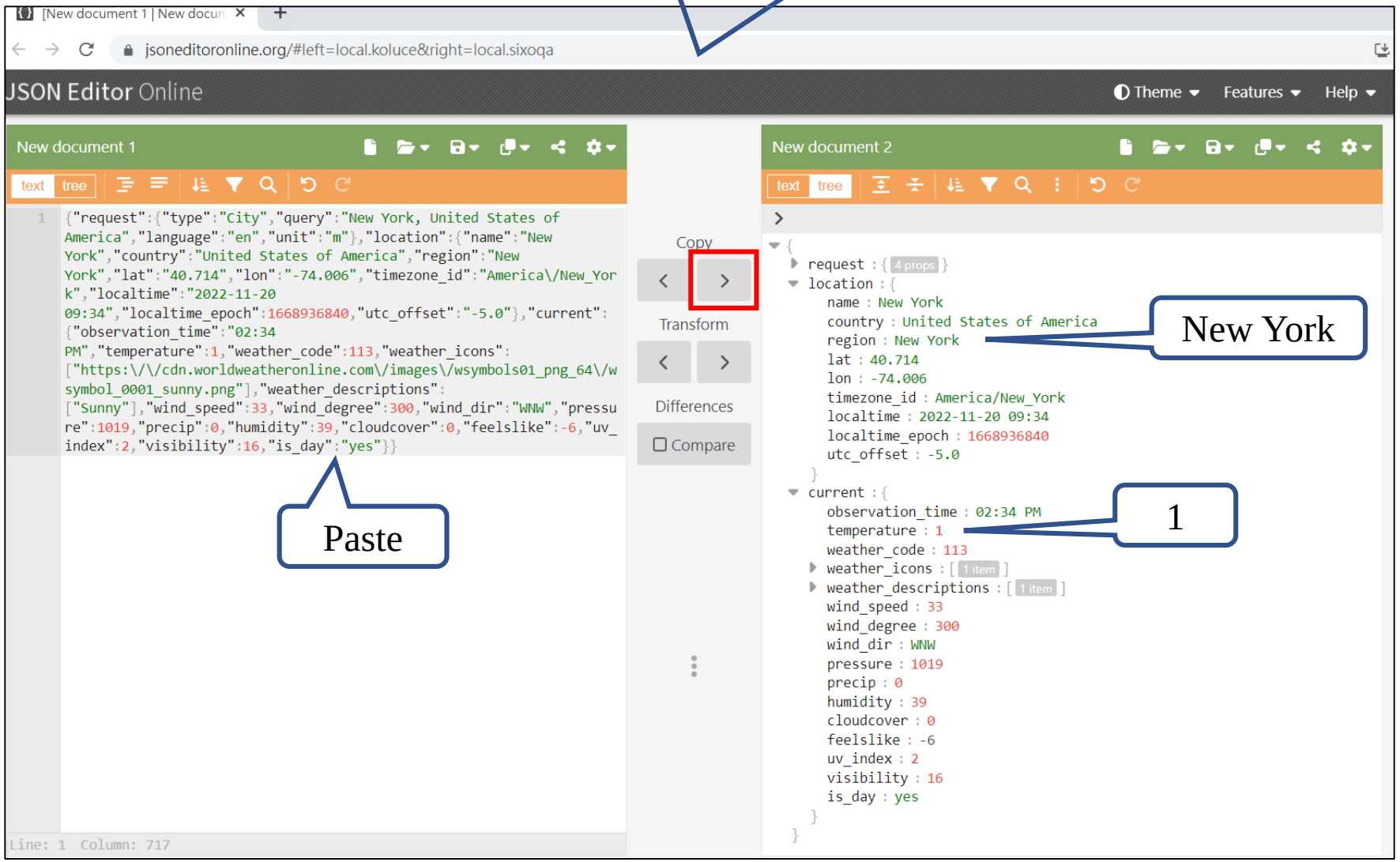



click
timestamp
http request
msg.payload
all nodes
11/20/2022, 10:36:02 PM node: 63acc057.d4855
msg.payload : string[716]
"{"request":{"type":"City","query":"New York, United States of America","language":"en","unit":"m"},"location":{"name":"New York","country":"United States of America","region":"New York","lat":"40.714","lon":"-74.006","timezone_id":"America\/New_York","localtime":"2022-11-20 09:34","localtime_epoch":1668936840,"utc_offset":"-5.0"},"current":{"observation_time":"02:34 PM","temperature":1,"weather_code":113,"weather_icons":["https:\/\/cdn.worldweatheronline.com\/images\/wsymbols01_png_64\/wsymbol_0001_sunny.png"],"weather_descriptions":["Sunny"],"wind_speed":33,"wind_degree":300,"wind_dir":"WNW","pressure":1019,"precip":0,"humidity":39,"cloudcover":0,"feelslike":-6,"uv_index":2,"visibility":16,"is_day":"yes"}}"

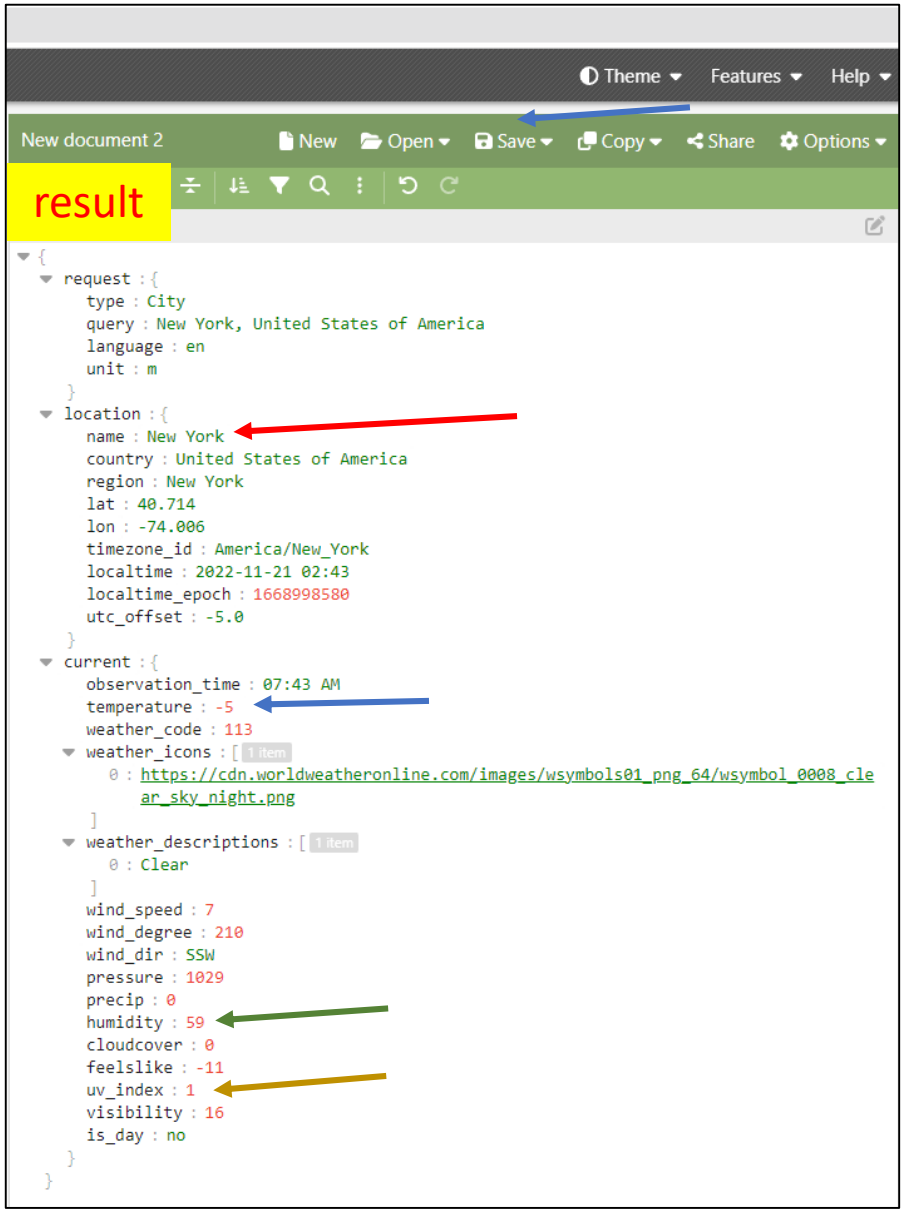
Copy

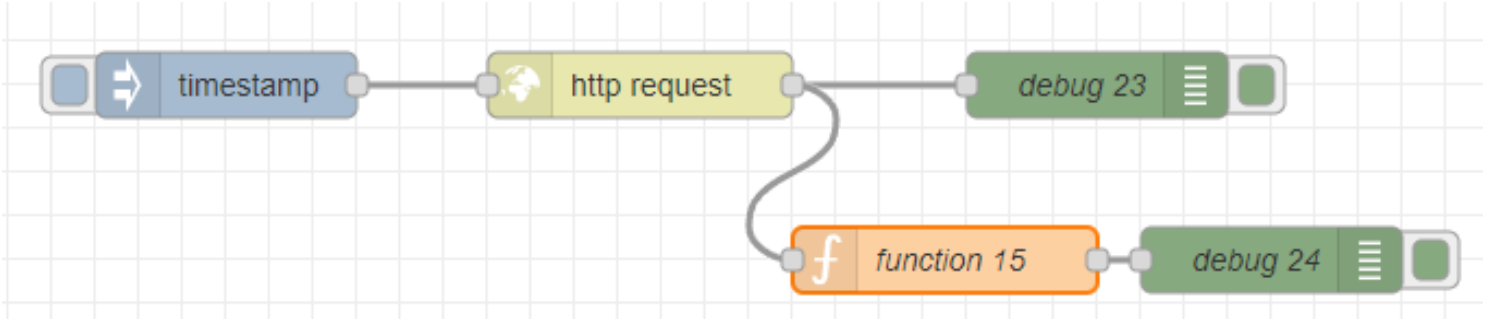

11/20/2022, 10:36:02 PM node: 63acc057.d4855
msg.payload : string[716]
"{"request":{"type":"City","query":"New York, United States of America","language":"en","unit":"m"},"location":{"name":"New York","country":"United States of America","region":"New York","lat":"40.714","lon":"-74.006","timezone_id":"America\/New_York","localtime":"2022-11-20 09:34","localtime_epoch":1668936840,"utc_offset":"-5.0"},"current":{"observation_time":"02:34 PM","temperature":1,"weather_code":113,"weather_icons":["https:\/\/cdn.worldweatheronline.com\/images\/wsymbols01_png_64\/wsymbol_0001_sunny.png"],"weather_descriptions":["Sunny"],"wind_speed":33,"wind_degree":300,"wind_dir":"WNW","pressure":1019,"precip":0,"humidity":39,"cloudcover":0,"feelslike":-6,"uv_index":2,"visibility":16,"is_day":"yes"}}"
copy

https://jsoneditoronline.org



- result.location.name
- result.current.temperature
- result.current.humidity
- result.current.uv_index





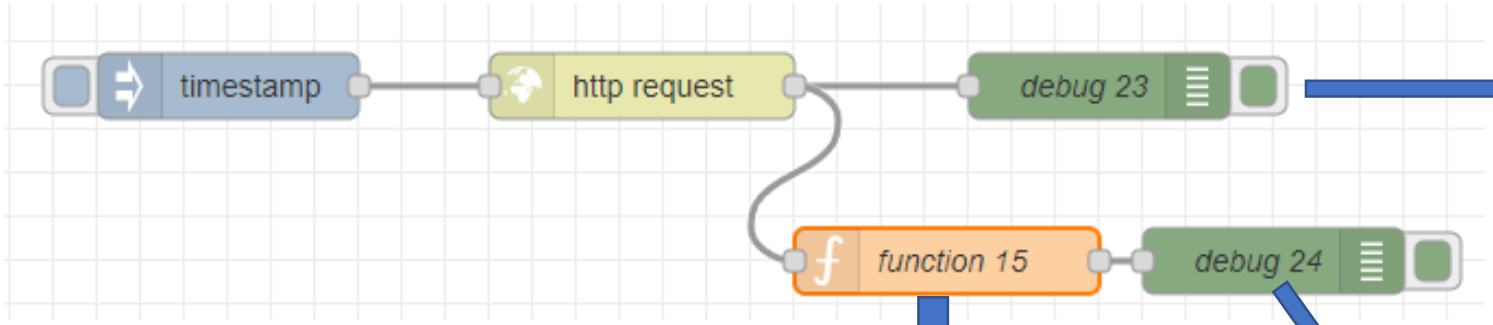
Triger

Get data using weatherstack API

JSON string to JSON object

Extract name, temperature, humidity, uv_index

Output
{temp: temperature,
name: name,
hum: humidity,
uv: uv_index
}



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

```
1
2   var result = JSON.parse(msg.payload);
3   var temp = result.current.temperature;
4   var name = result.location.name;
5   var hum = result.current.humidity;
6   var uv = result.current.uv_index;
7   msg.payload = { temp: temp, name: name, hum: hum, uv: uv};
8   return msg;
```

debug

all nodes | all

11/22/2022, 1:40:06 PM node: debug 23

msg.payload : string[725]


```
"{"request":{"type":"City","query":"New York, United States of America","language":"en","unit":"m"},"location":{"name":"New York","country":"United States of America","region":"New York","lat":"40.714","lon":"-74.006","timezone_id":"America\/New_York","localtime":"2022-11-22 00:35","localtime_epoch":1669077300,"utc_offset":"-5.0"},"current":{"observation_time":"05:35 AM","temperature":2,"weather_code":113,"weather_icons":["https:\/\/cdn.worldweatheronline.com\/images\/wsymbols01_png_64\/wsymbol_0008_clear_sky_night.png"],"weather_descriptions":["Clear"],"wind_speed":13,"wind_degree":240,"wind_dir":"WSW","pressure":1025,"precip":0,"humidity":56,"cloudcover":0,"feelslike":-3,"uv_index":1,"visibility":16,"is_day":"no"}}"
```


11/22/2022, 1:40:06 PM node: debug 24

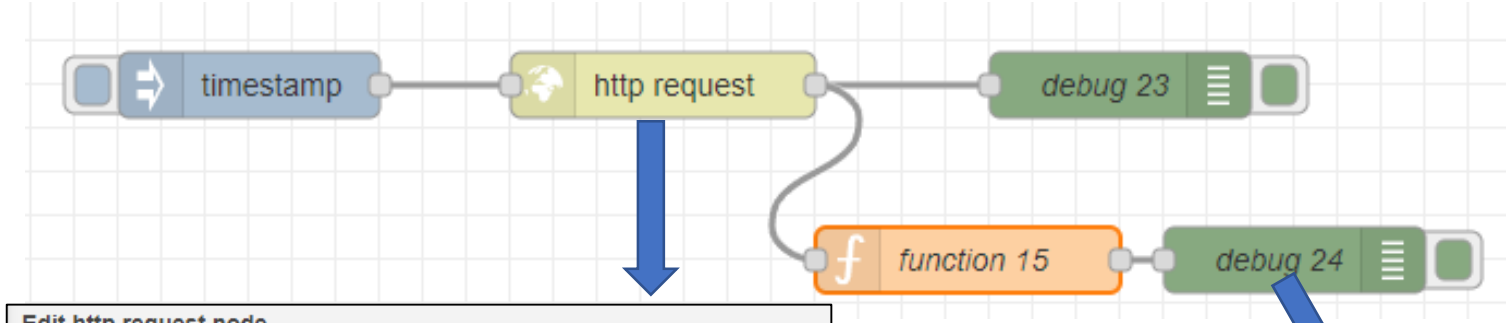
msg.payload : Object

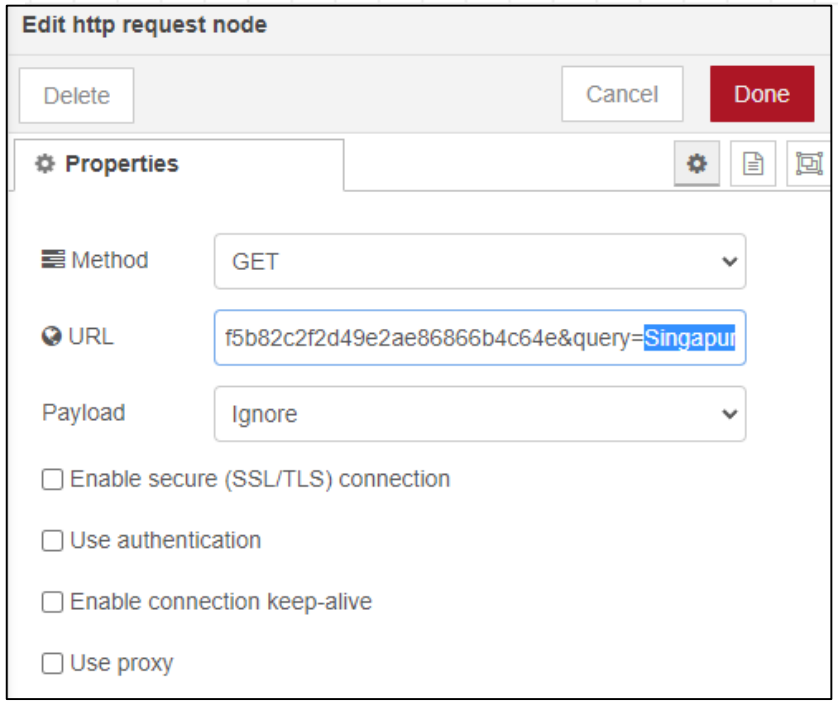

```
{ temp: 2, name: "New York", hum: 56, uv: 1 }
```

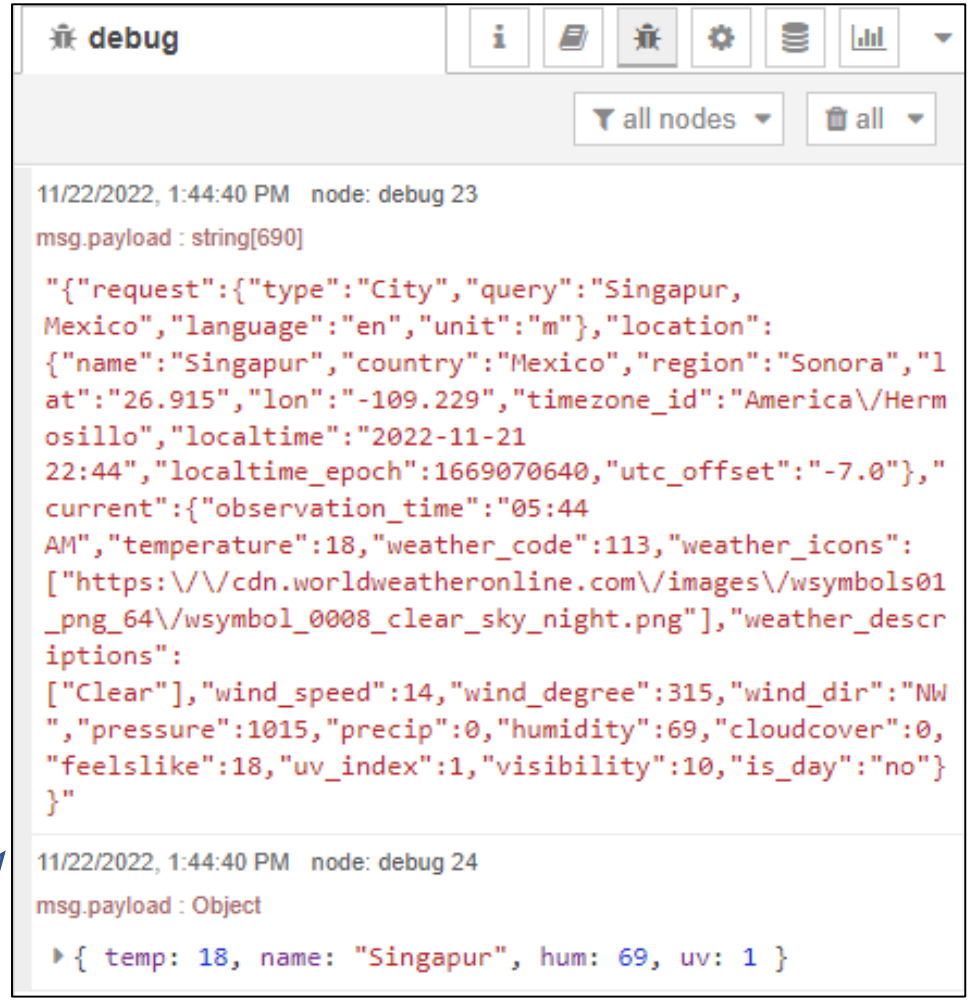
Exercise 10-1: Check seven cities weather

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

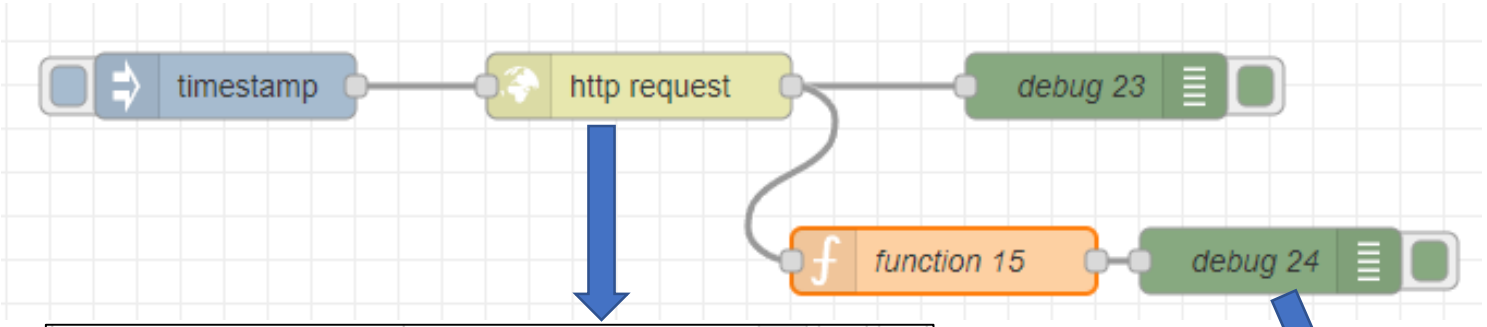
Singapur

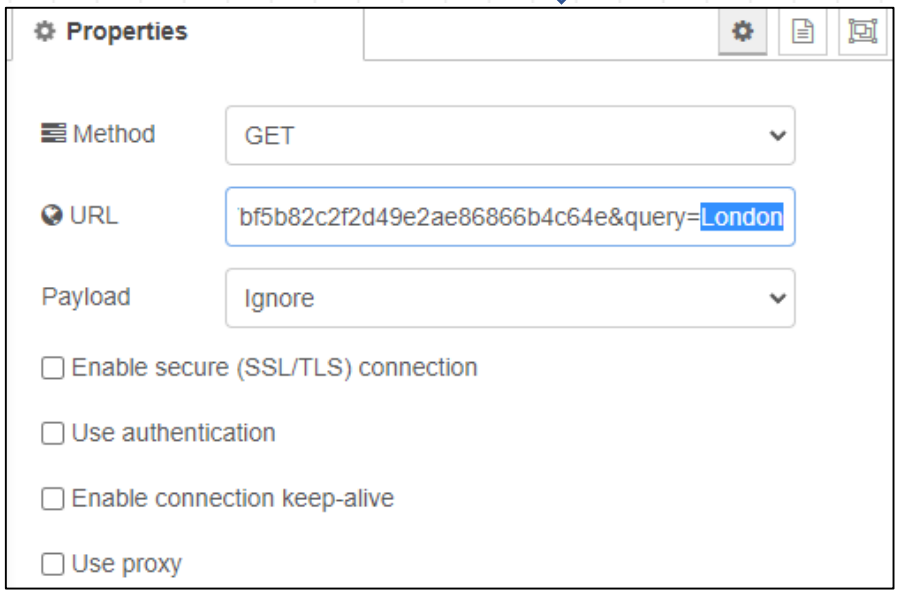

timestamp
http request
debug 23
function 15
debug 24

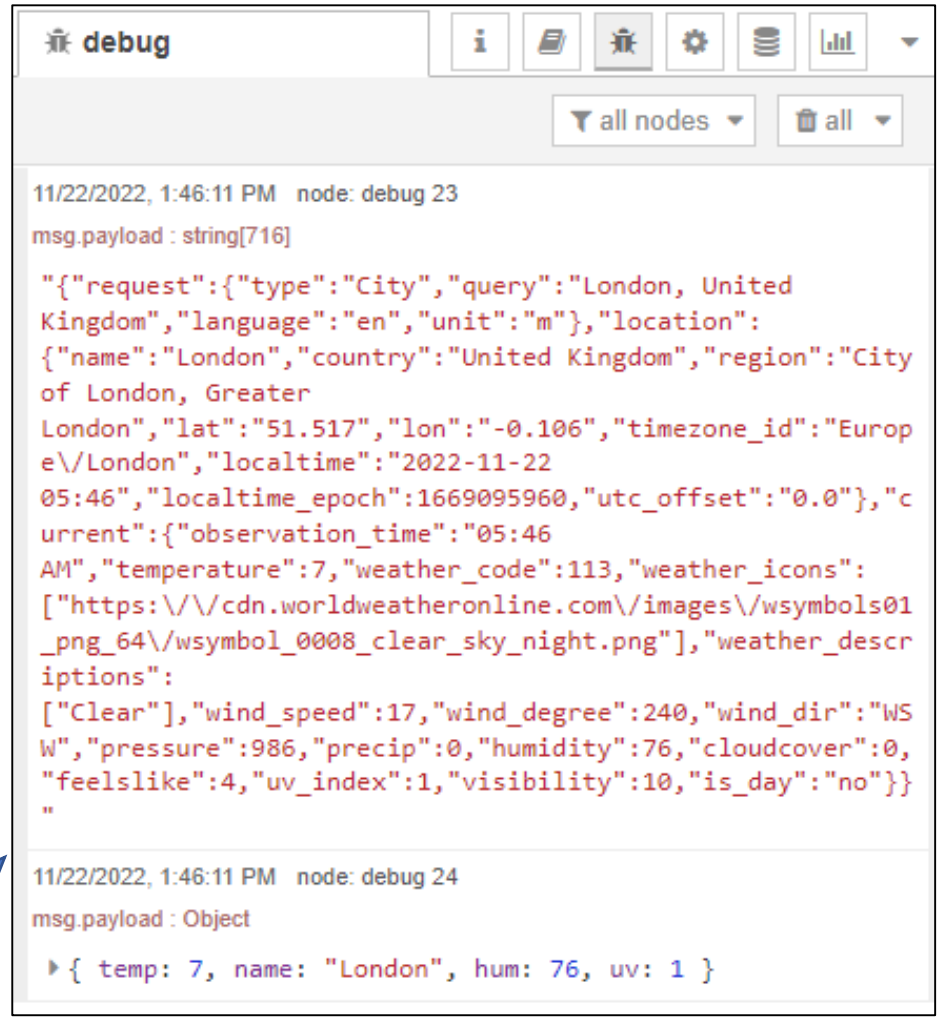

Edit http request node
Delete
Cancel
Done
Properties
Method
GET
URL
f5b82c2f2d49e2ae86866b4c64e&query=Singapur
Payload
Ignore
Enable secure (SSL/TLS) connection
Use authentication
Enable connection keep-alive
Use proxy


debug
all nodes
all
11/22/2022, 1:44:40 PM node: debug 23
msg.payload : string[690]
"{"request":{"type":"City","query":"Singapur, Mexico","language":"en","unit":"m"},"location":{"name":"Singapur","country":"Mexico","region":"Sonora","lat":"26.915","lon":"-109.229","timezone_id":"America\/Hermosillo","localtime":"2022-11-21 22:44","localtime_epoch":1669070640,"utc_offset":"-7.0"},"current":{"observation_time":"05:44 AM","temperature":18,"weather_code":113,"weather_icons":["https:\/\/cdn.worldweatheronline.com\/images\/wsymbols01_png_64\/wsymbol_0008_clear_sky_night.png"],"weather_descriptions":["Clear"],"wind_speed":14,"wind_degree":315,"wind_dir":"NW","pressure":1015,"precip":0,"humidity":69,"cloudcover":0,"feelslike":18,"uv_index":1,"visibility":10,"is_day":"no"}}"
11/22/2022, 1:44:40 PM node: debug 24
msg.payload : Object
{ temp: 18, name: "Singapur", hum: 69, uv: 1 }

London


timestamp
http request
debug 23
function 15
debug 24

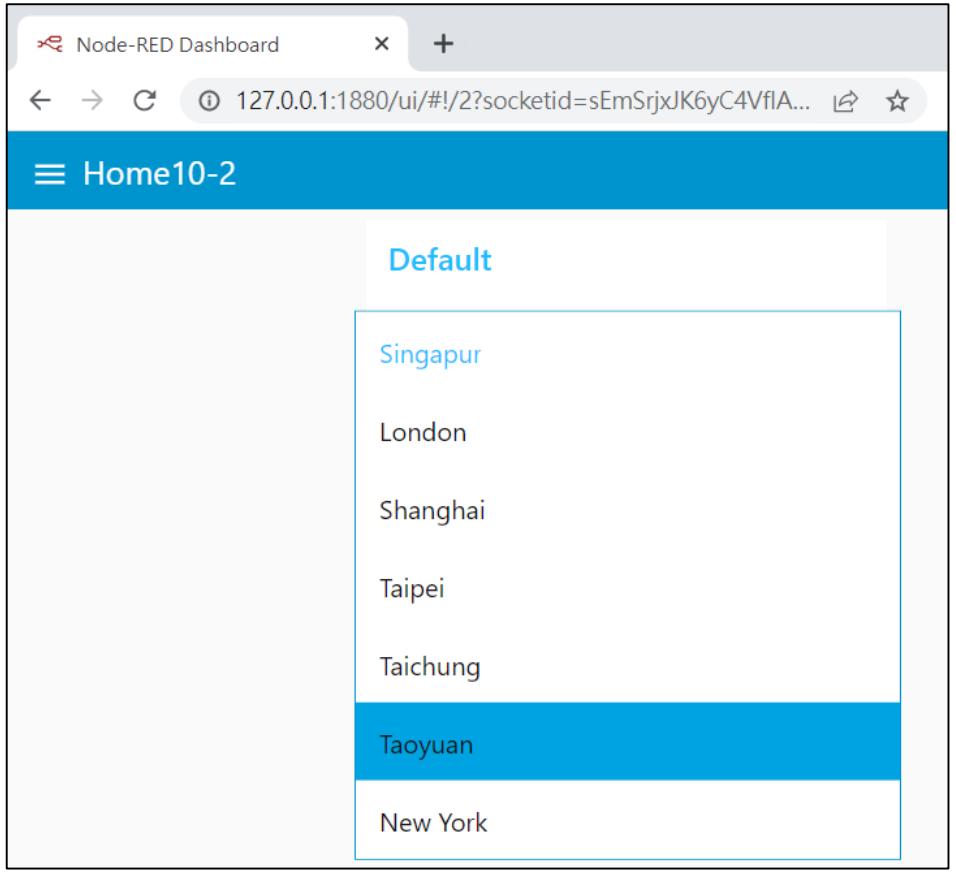

Properties
Method
GET
URL
bf5b82c2f2d49e2ae86866b4c64e&query=London
Payload
Ignore
Enable secure (SSL/TLS) connection
Use authentication
Enable connection keep-alive
Use proxy


debug
all nodes
all
11/22/2022, 1:46:11 PM node: debug 23
msg.payload : string[716]
"{"request":{"type":"City","query":"London, United Kingdom","language":"en","unit":"m"},"location":{"name":"London","country":"United Kingdom","region":"City of London, Greater London","lat":"51.517","lon":"-0.106","timezone_id":"Europe\/London","localtime":"2022-11-22 05:46","localtime_epoch":1669095960,"utc_offset":"0.0"},"current":{"observation_time":"05:46 AM","temperature":7,"weather_code":113,"weather_icons":["https:\/\/cdn.worldweatheronline.com\/images\/wsymbols01_png_64\/wsymbol_0008_clear_sky_night.png"],"weather_descriptions":["Clear"],"wind_speed":17,"wind_degree":240,"wind_dir":"WSW","pressure":986,"precip":0,"humidity":76,"cloudcover":0,"feelslike":4,"uv_index":1,"visibility":10,"is_day":"no"}}"
11/22/2022, 1:46:11 PM node: debug 24
msg.payload : Object
{ temp: 7, name: "London", hum: 76, uv: 1 }

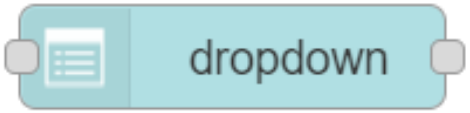
Exercise 10-2

- Using dropdown to make a interface for city selection.

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



dropdown



dropdown

Adds a dropdown select box to the user interface.

Multiple value / label pairs can be added as required. If the label is not specified the value will be used for both.

The configured value of the selected item will be returned as `msg.payload`.

Setting `msg.payload` to one of the item values will preset the choice in the dropdown. If using the multi-select option then the payload should be an array of values.

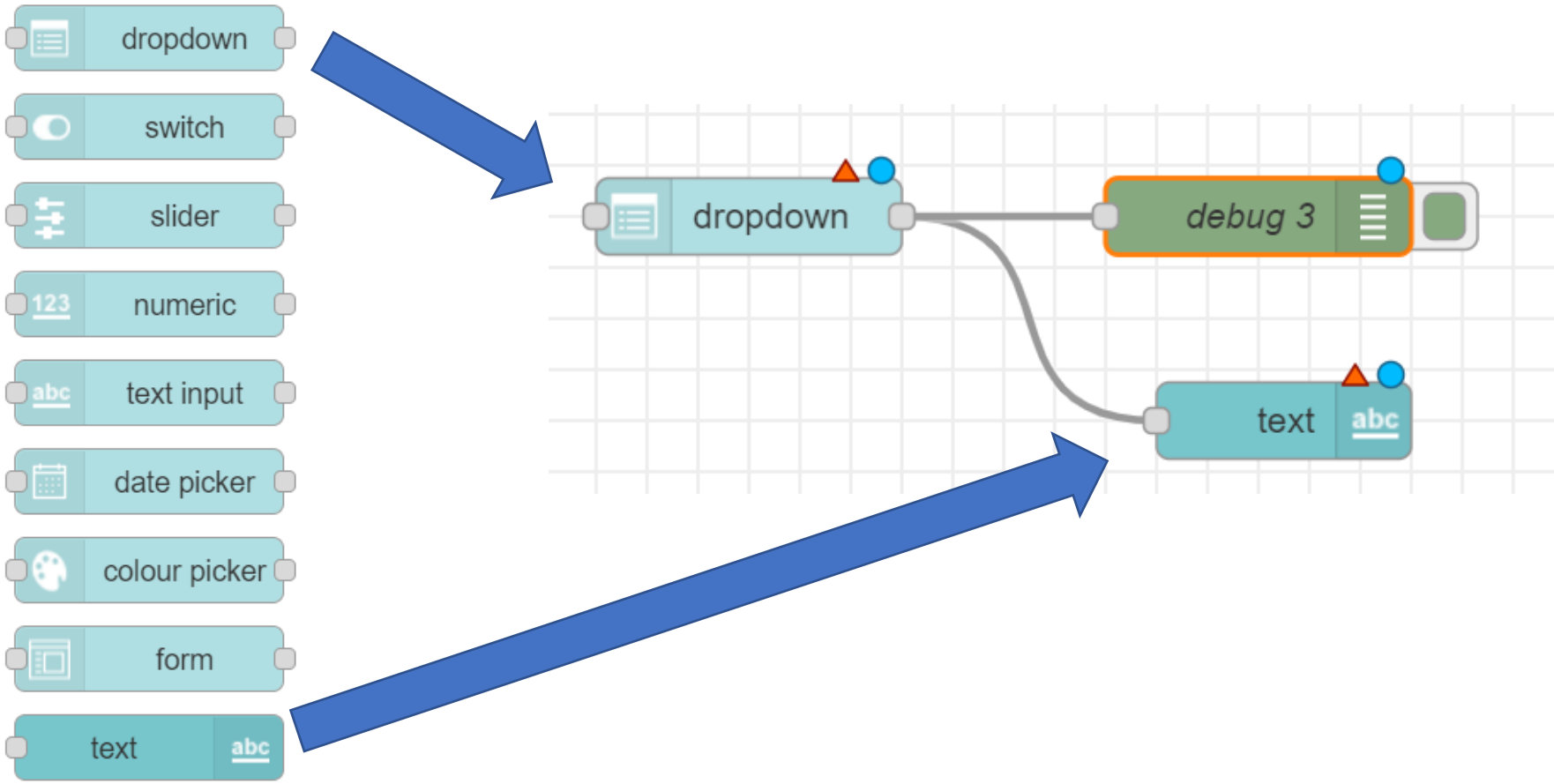
Optionally the **Topic** field can be used to set the `msg.topic` property.

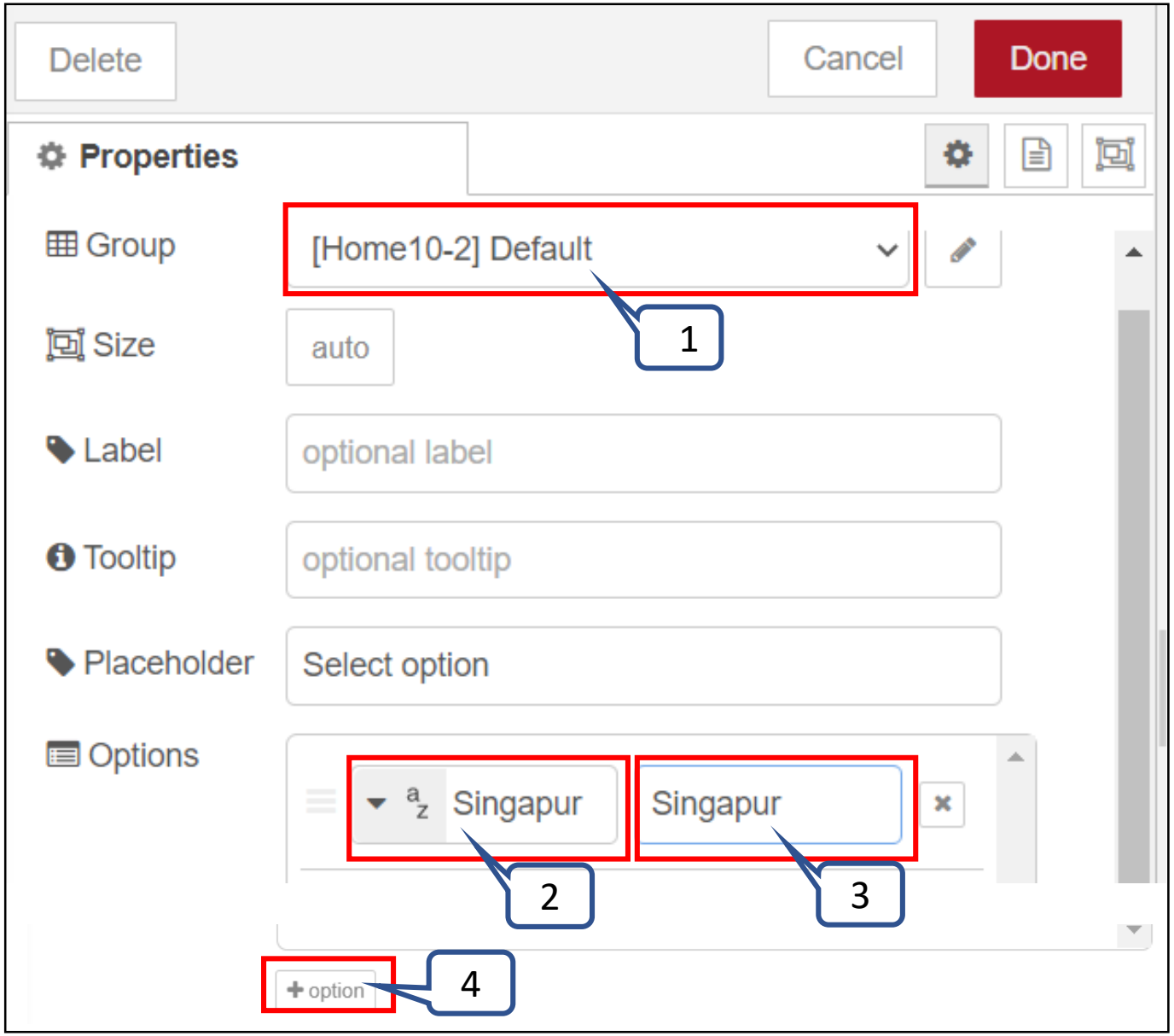
The Options may be configured by inputting `msg.options` containing an array. If just text then the value will be the same as the label, otherwise you can specify both by using an object of `"label":"value"` pairs :

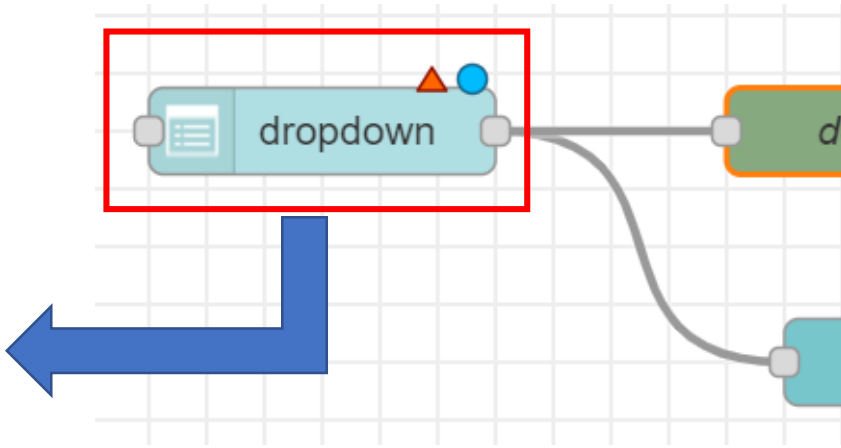
```
[ "Choice 1", "Choice 2", {"Choice 3":"3"} ]
```

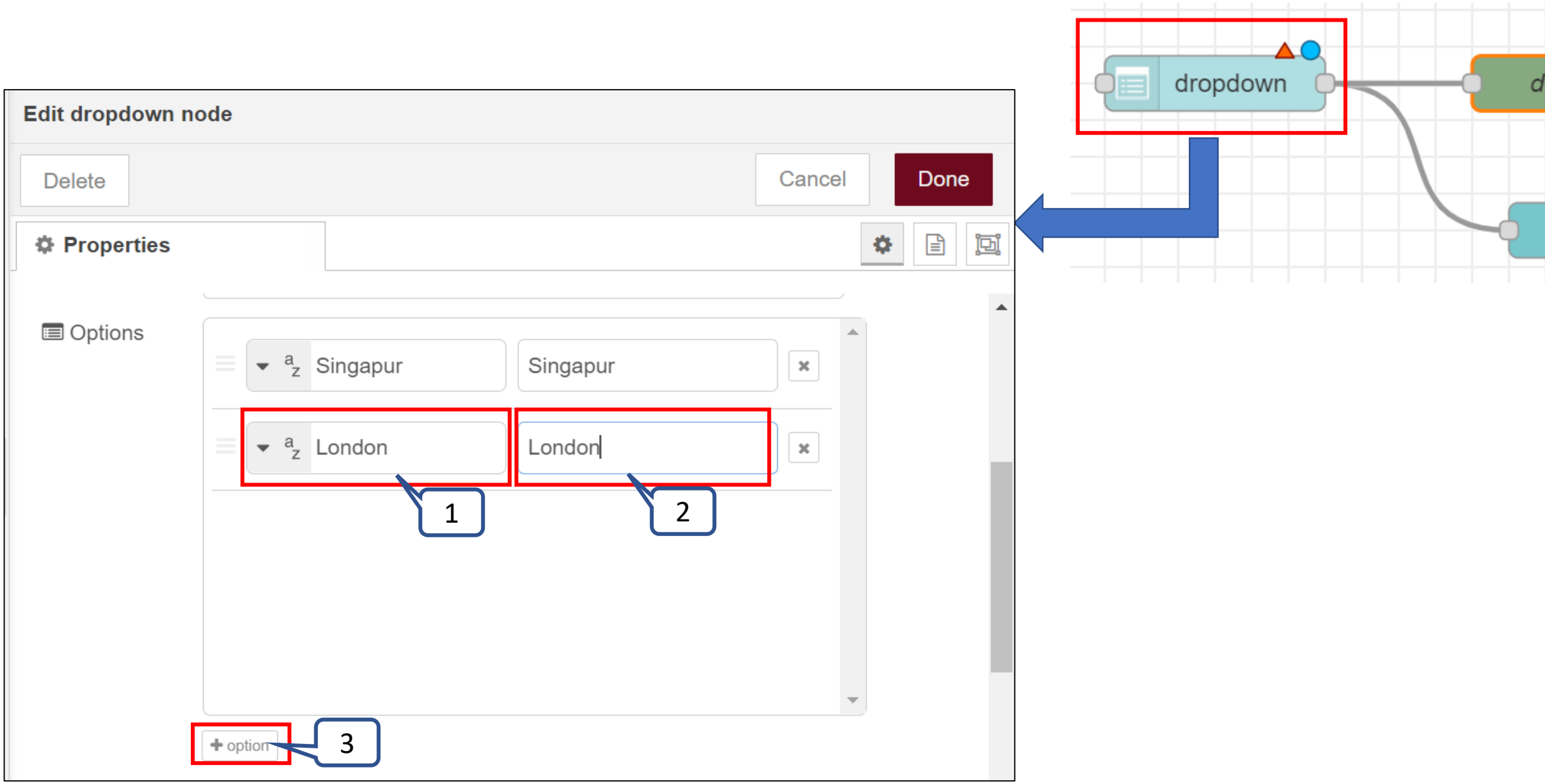
If the "Allow multiple selections" output option is enabled - the result will be returned as an array instead of a string.

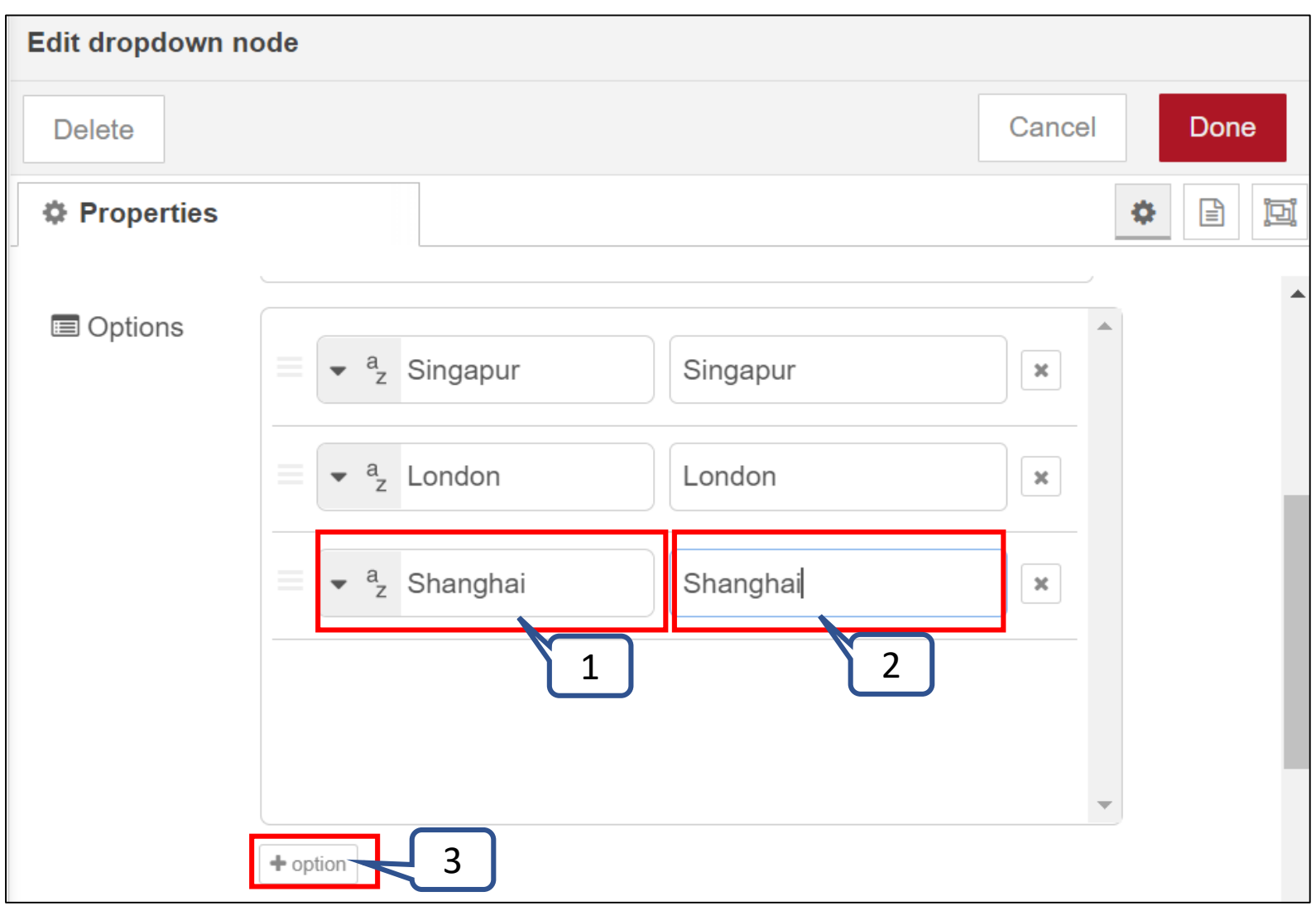
If a **Class** is specified, it will be added to the parent card. This way you can style the card and the elements inside it with custom CSS. The Class can be set at runtime by setting a `msg.className` string property.

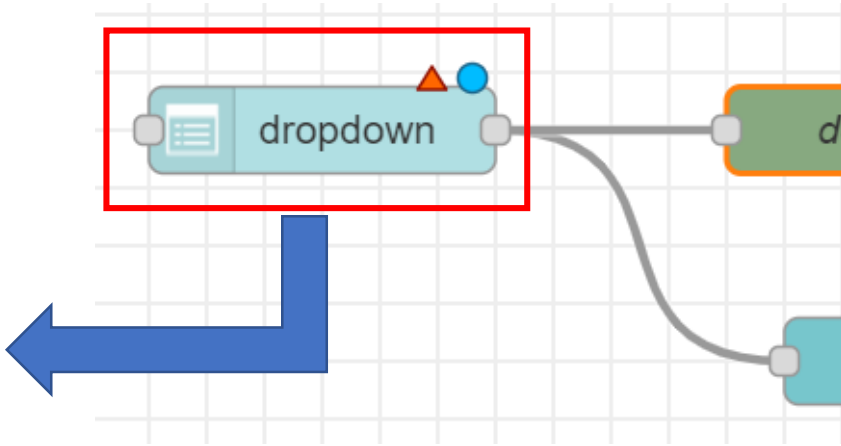

dropdown
switch
slider
123
numeric
abc
text input
date picker
colour picker
form
text
abc
dropdown
debug 3
text
abc

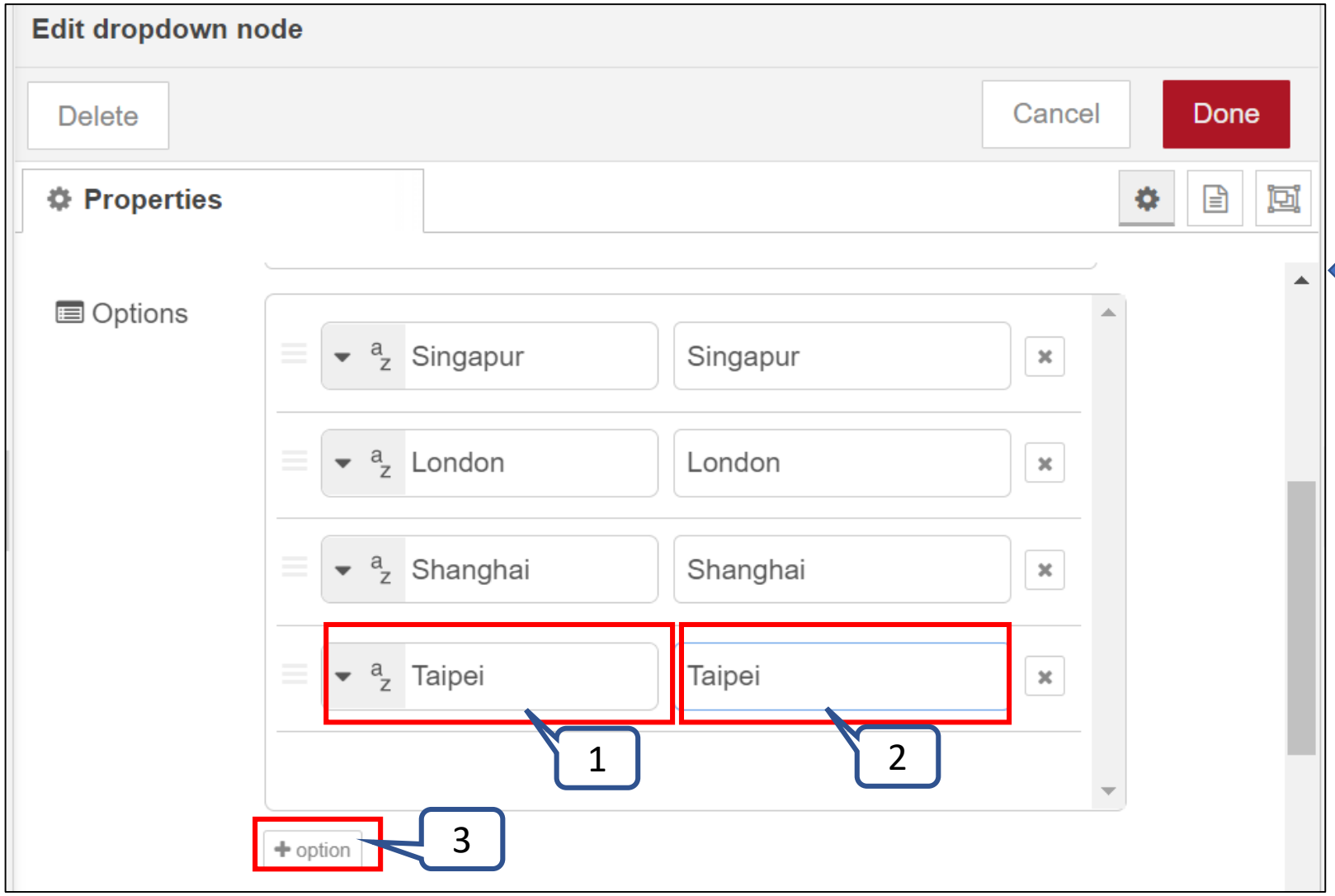

Delete
Cancel
Done
Properties
Group
[Home10-2] Default
1
Size
auto
Label
optional label
Tooltip
optional tooltip
Placeholder
Select option
Options
Singapur
Singapur
2
3
option
4

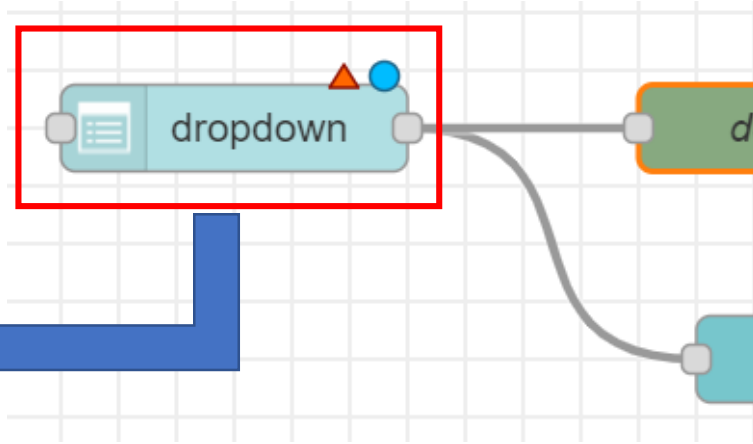

dropdown

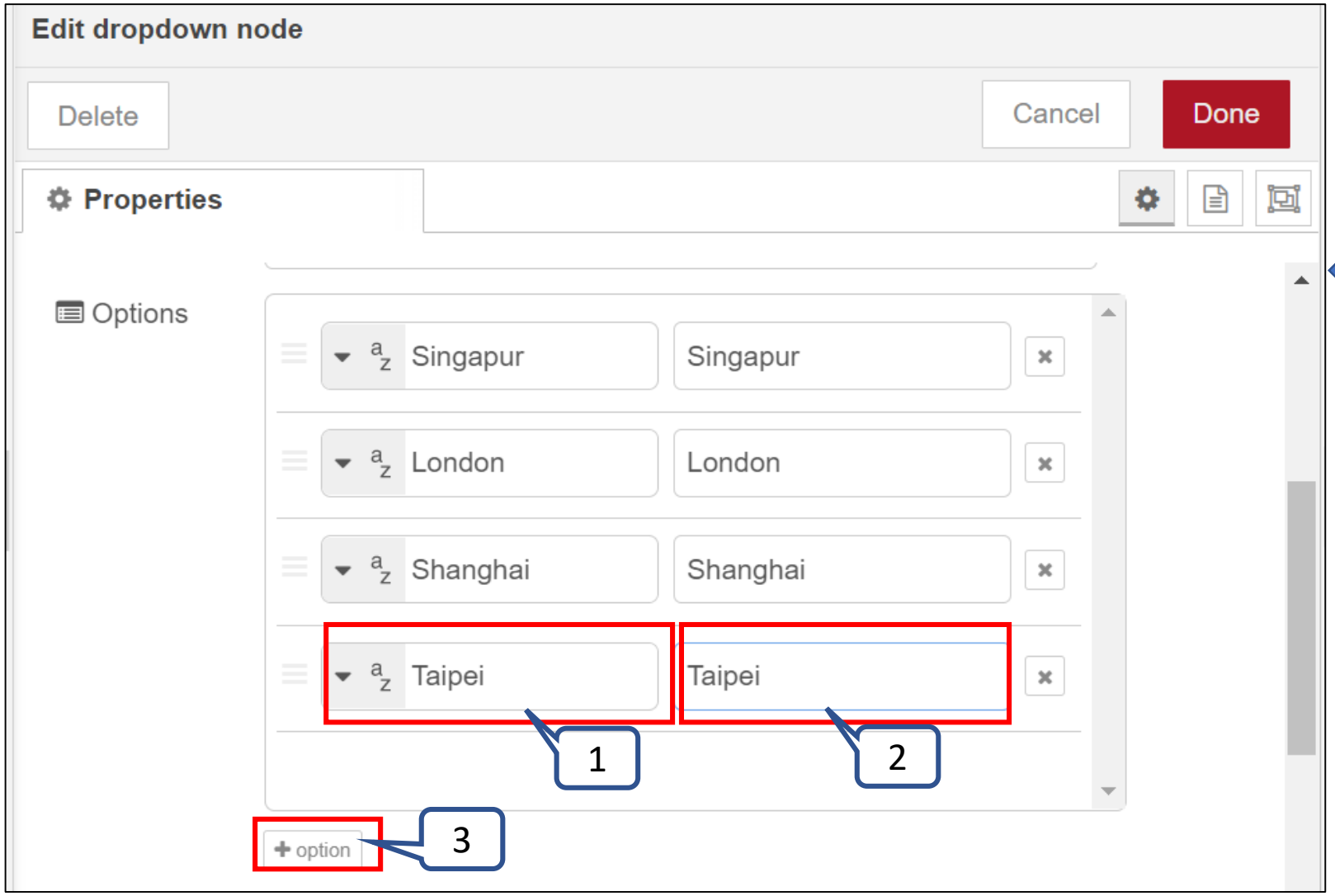

Edit dropdown node
Delete
Cancel
Done
Properties
Options
Singapur
Singapur
London
London
1
2
option
3
dropdown

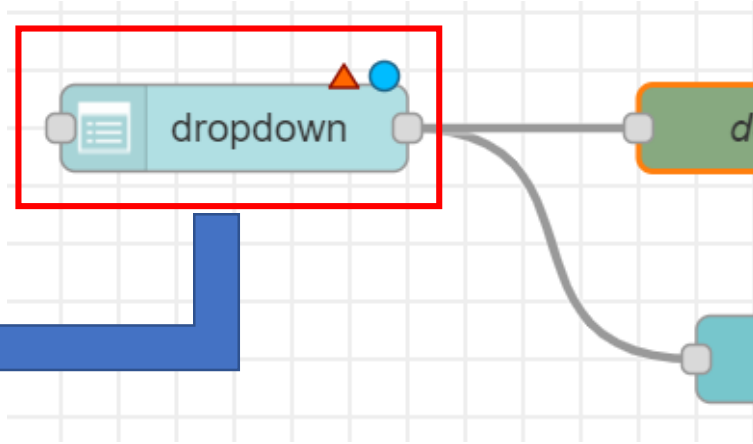

Edit dropdown node
Delete
Cancel
Done
Properties
Options
Singapur
Singapur
London
London
Shanghai
Shanghai
1
2
option
3

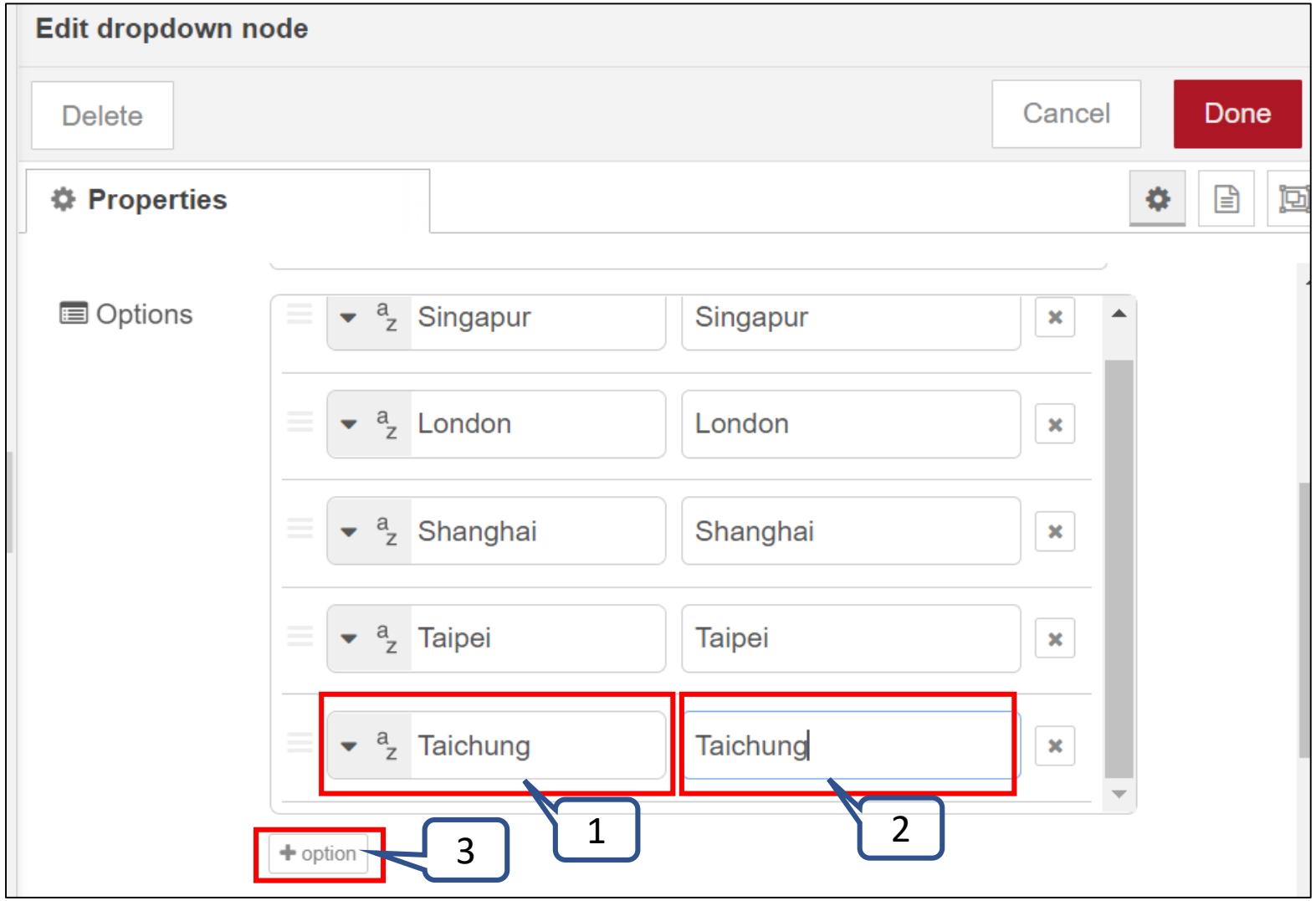

dropdown

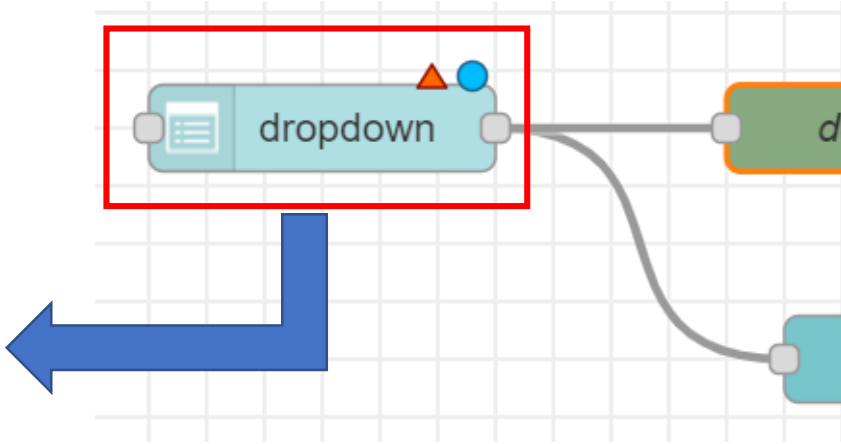

Edit dropdown node
Delete
Cancel
Done
Properties
Options
Singapur
Singapur
London
London
Shanghai
Shanghai
Taipei
Taipei
1
2
option
3

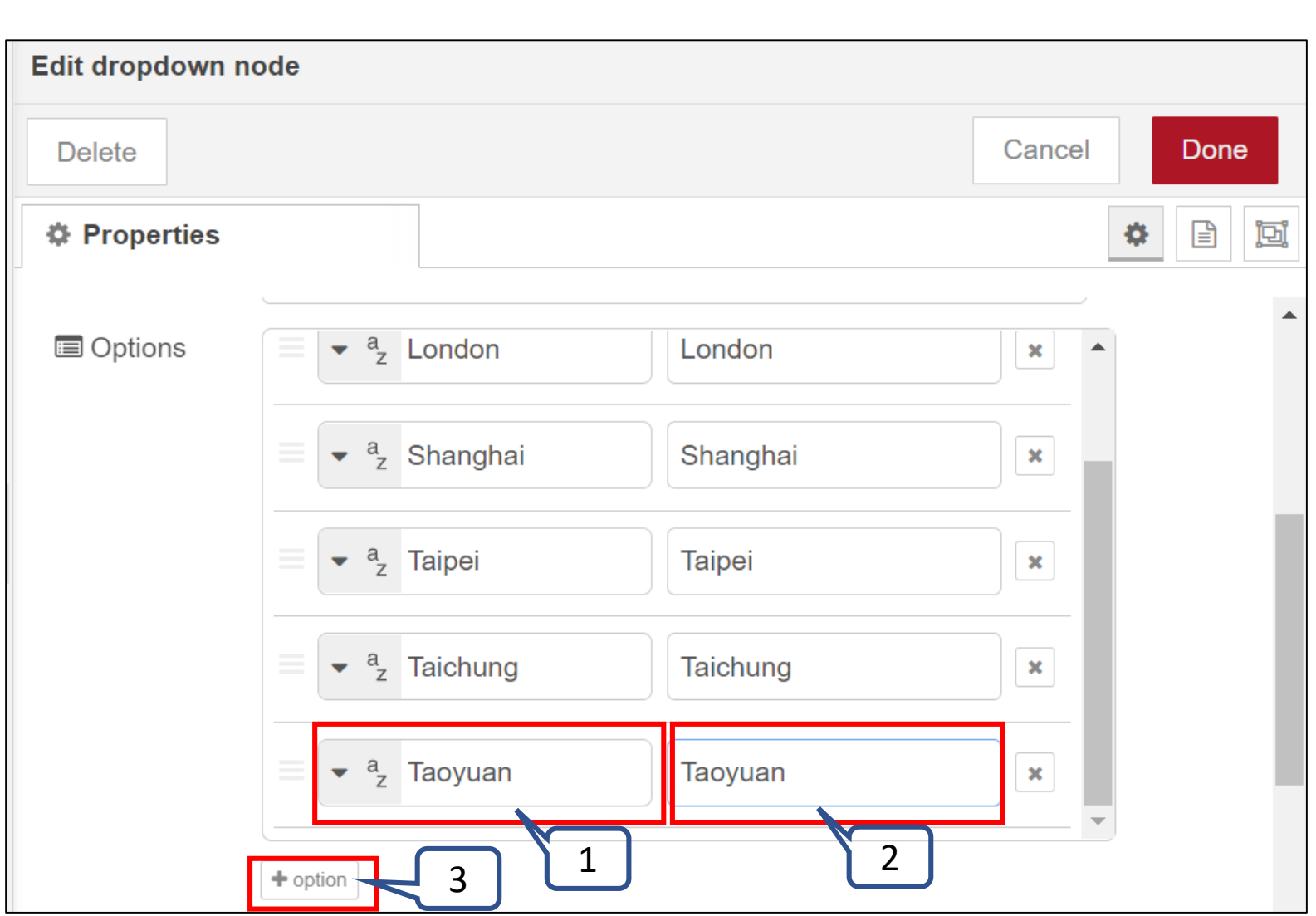

dropdown

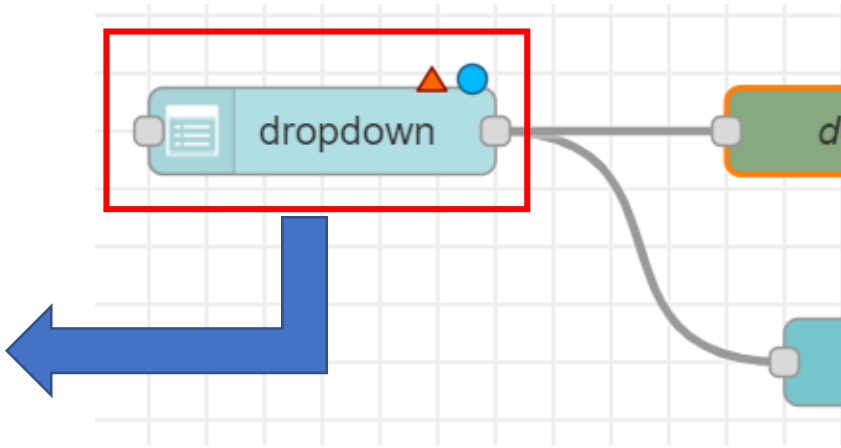

Edit dropdown node
Delete
Cancel
Done
Properties
Options
Singapur
Singapur
London
London
Shanghai
Shanghai
Taipei
Taipei
1
2
option
3

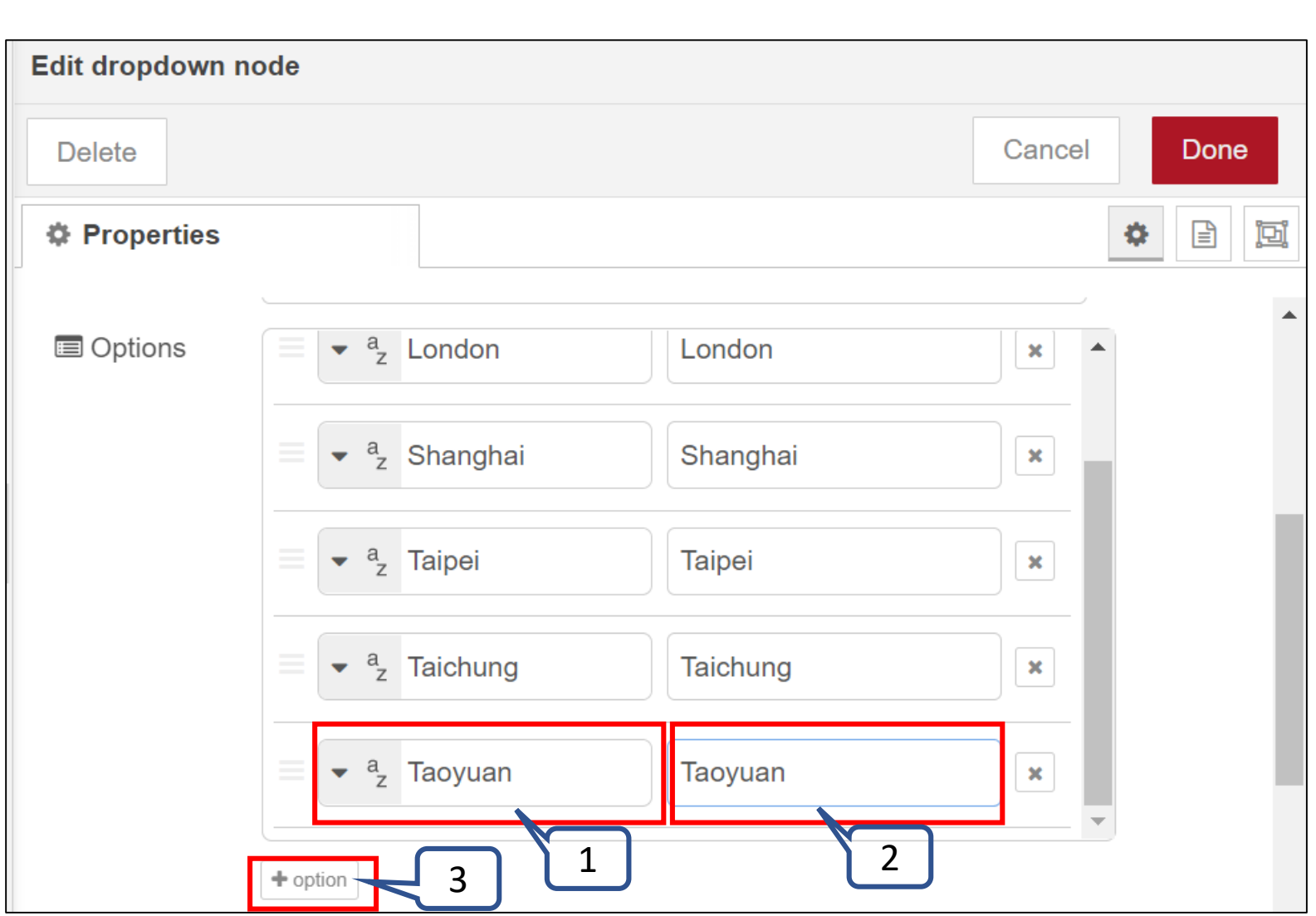

dropdown

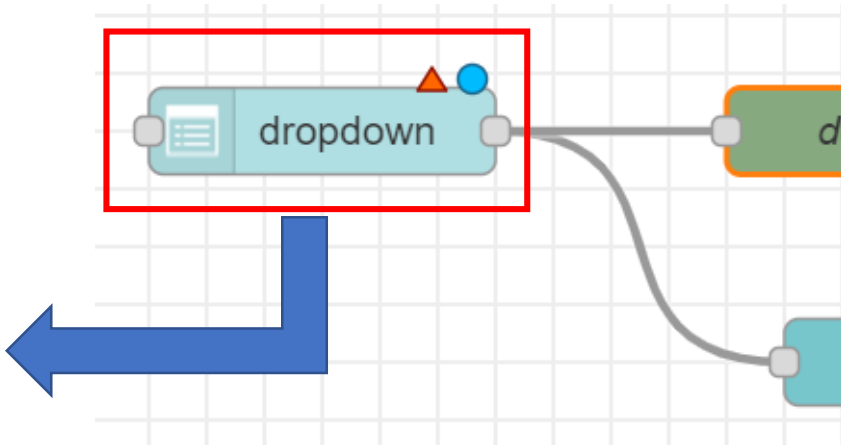

Edit dropdown node
Delete
Cancel
Done
Properties
Options
Singapur
Singapur
London
London
Shanghai
Shanghai
Taipei
Taipei
Taichung
Taichung
option
1
2
3

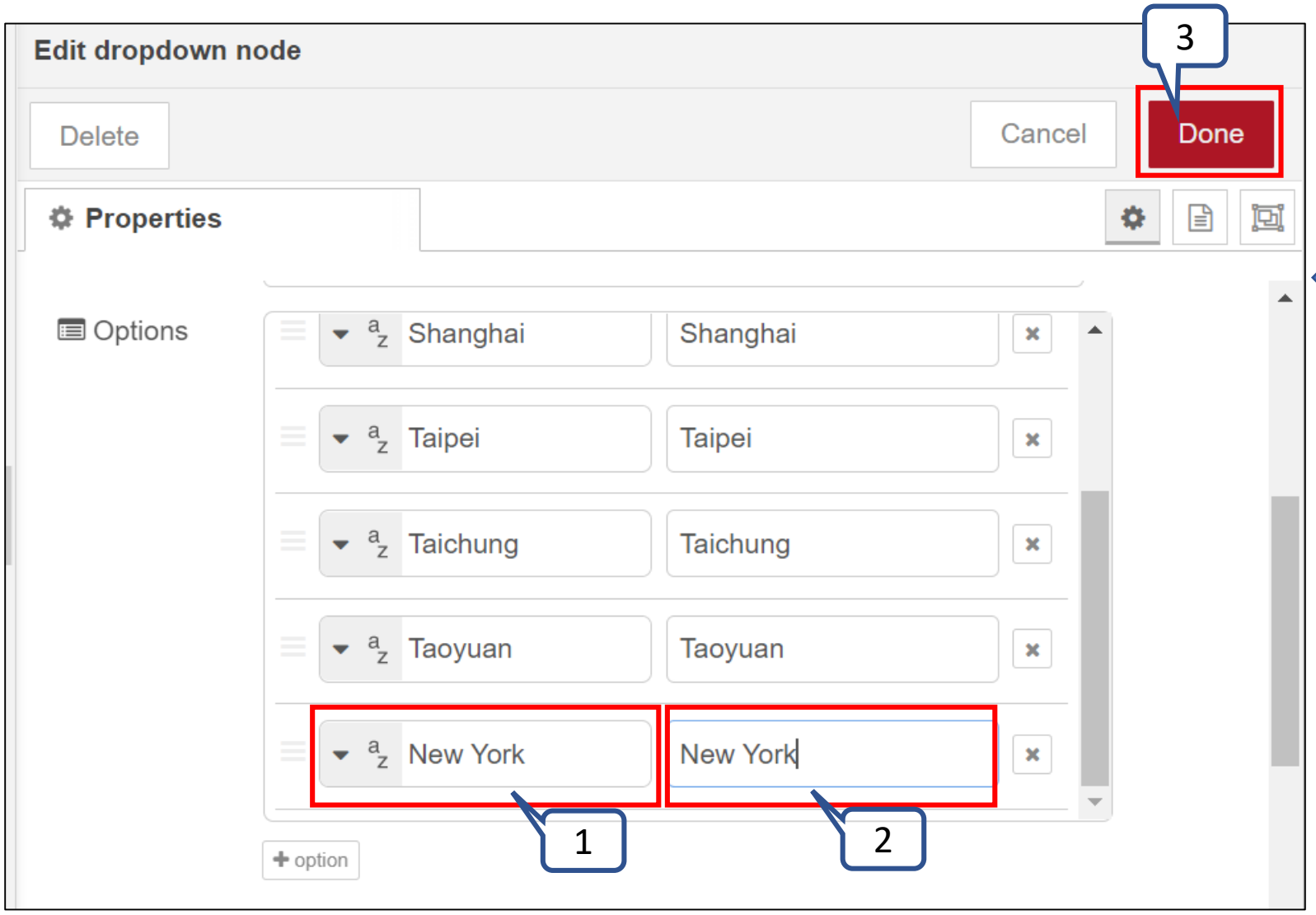

dropdown

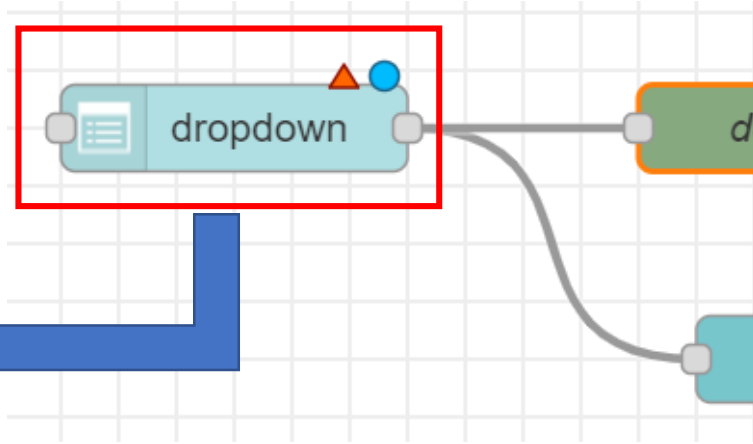

Edit dropdown node
Delete
Cancel
Done
Properties
Options
London
London
Shanghai
Shanghai
Taipei
Taipei
Taichung
Taichung
Taoyuan
Taoyuan
option
1
2
3

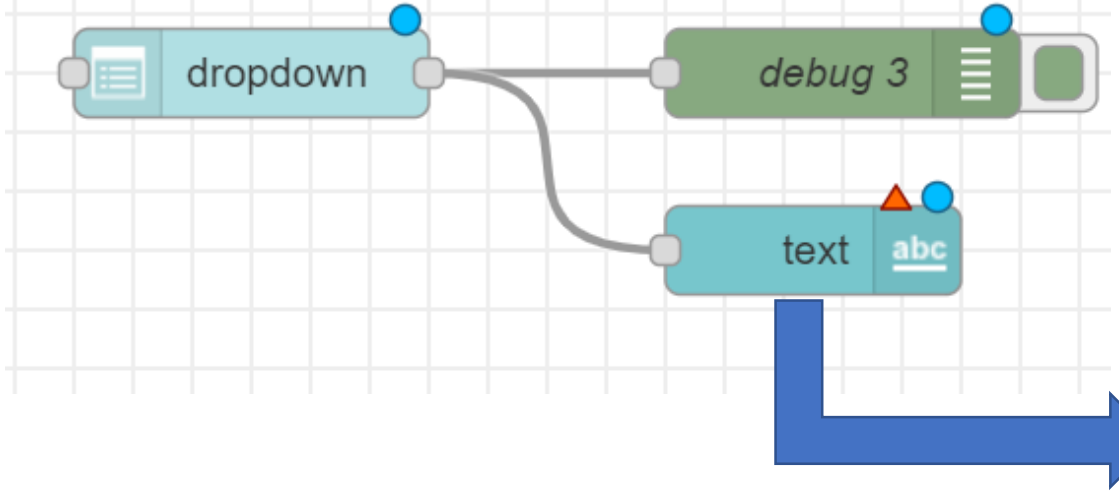

dropdown

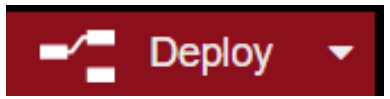

Edit dropdown node
Delete
Cancel
Done
Properties
Options
London
London
Shanghai
Shanghai
Taipei
Taipei
Taichung
Taichung
Taoyuan
Taoyuan
option
1
2
3

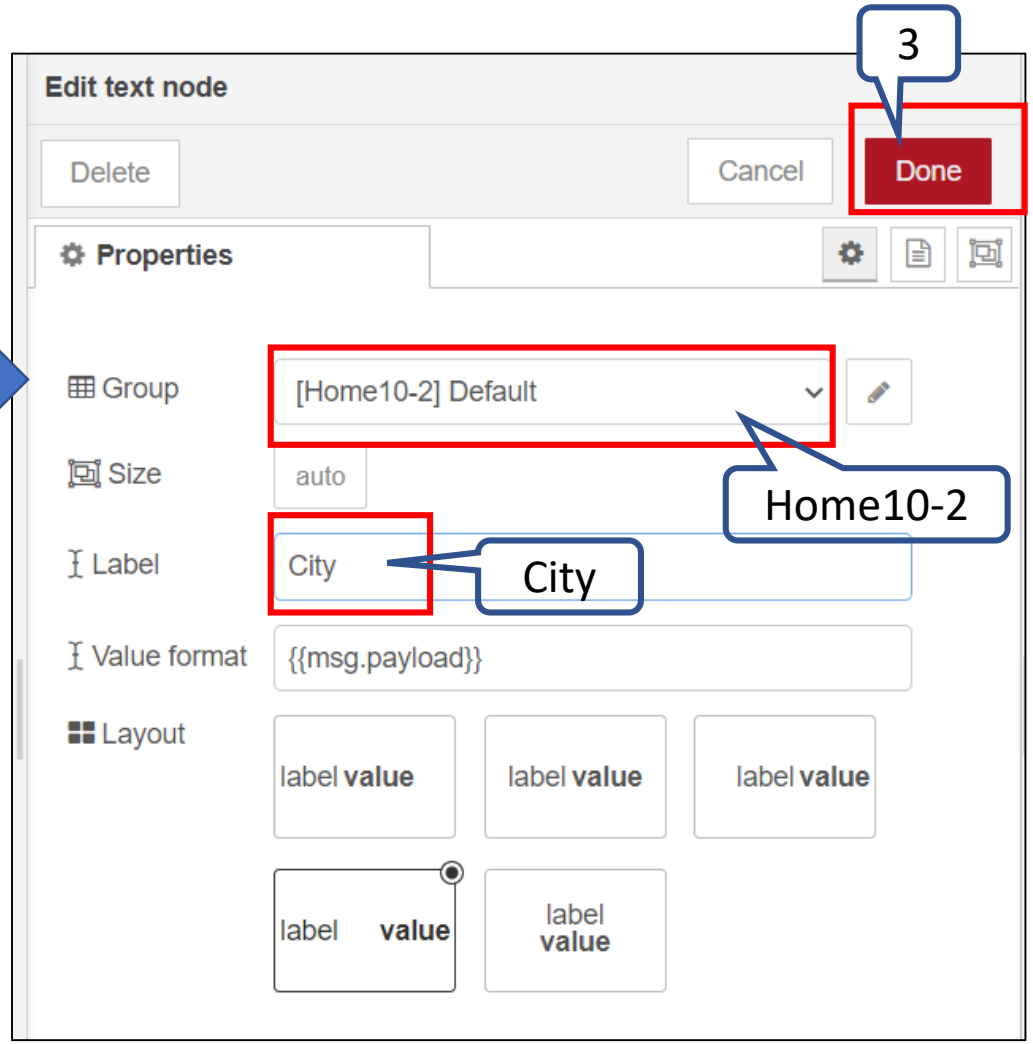

dropdown

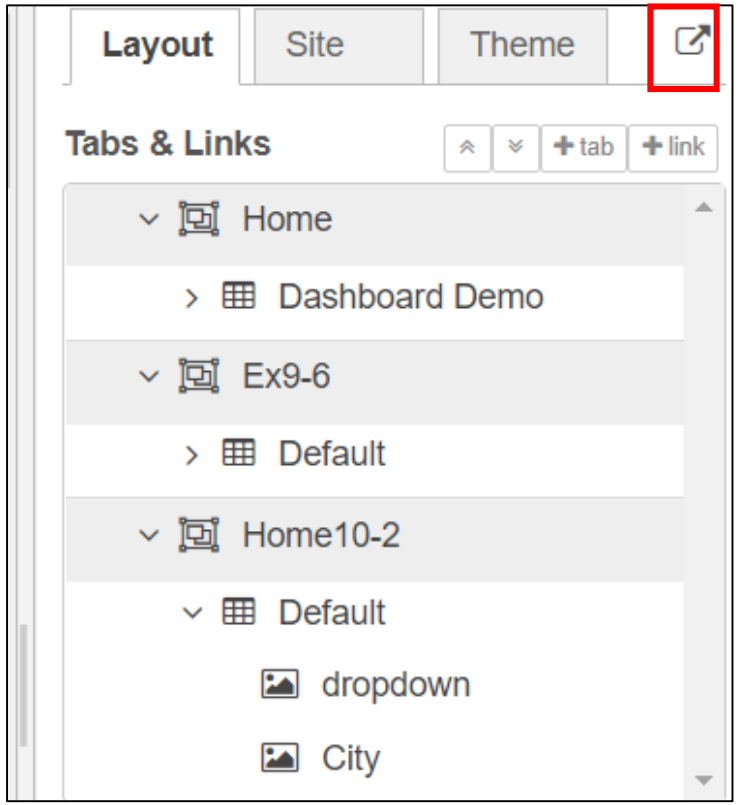

Edit dropdown node
Delete
Cancel
Done
3
Properties
Options
Shanghai
Shanghai
Taipei
Taipei
Taichung
Taichung
Taoyuan
Taoyuan
New York
New York
1
2
option

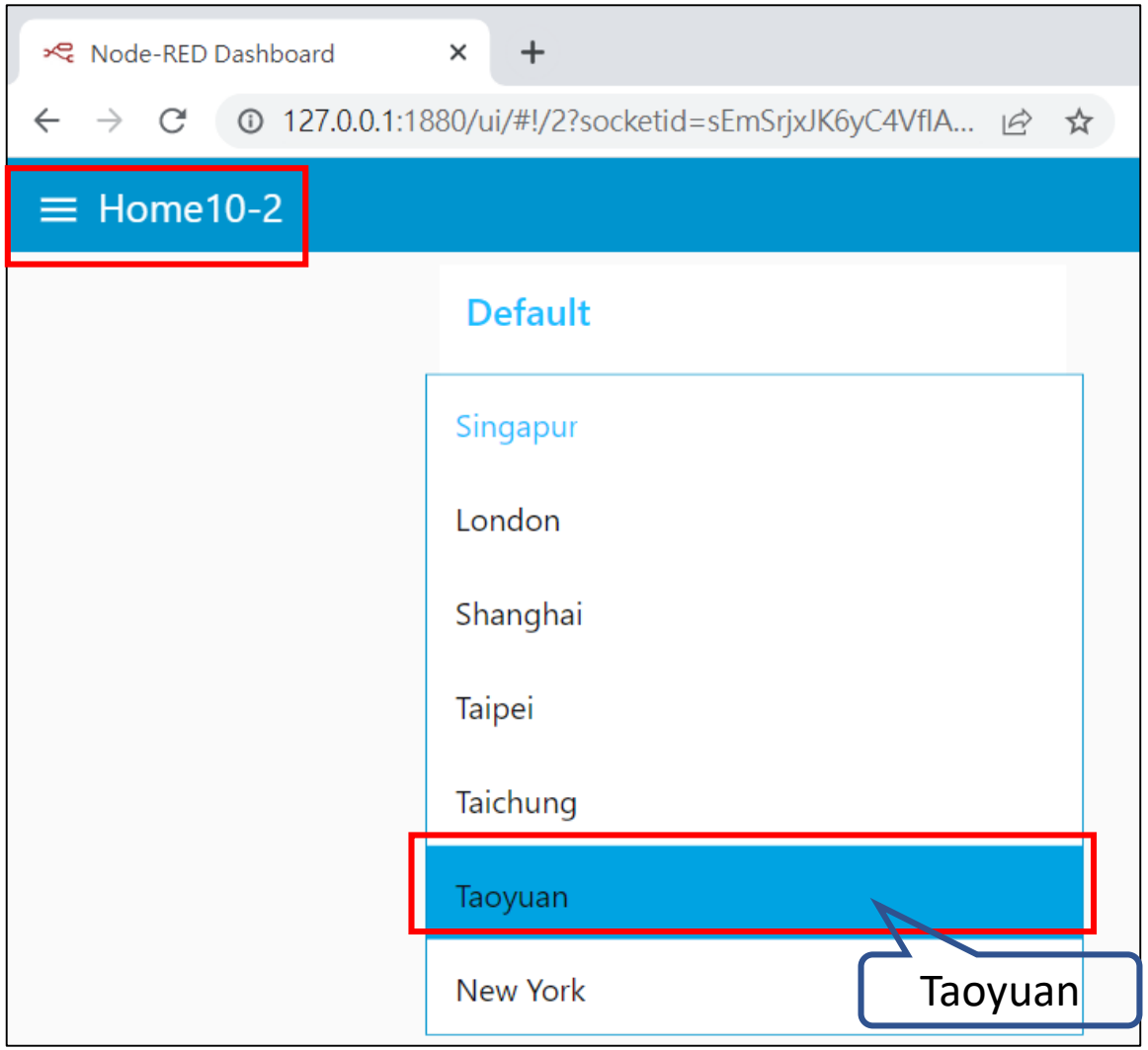

dropdown

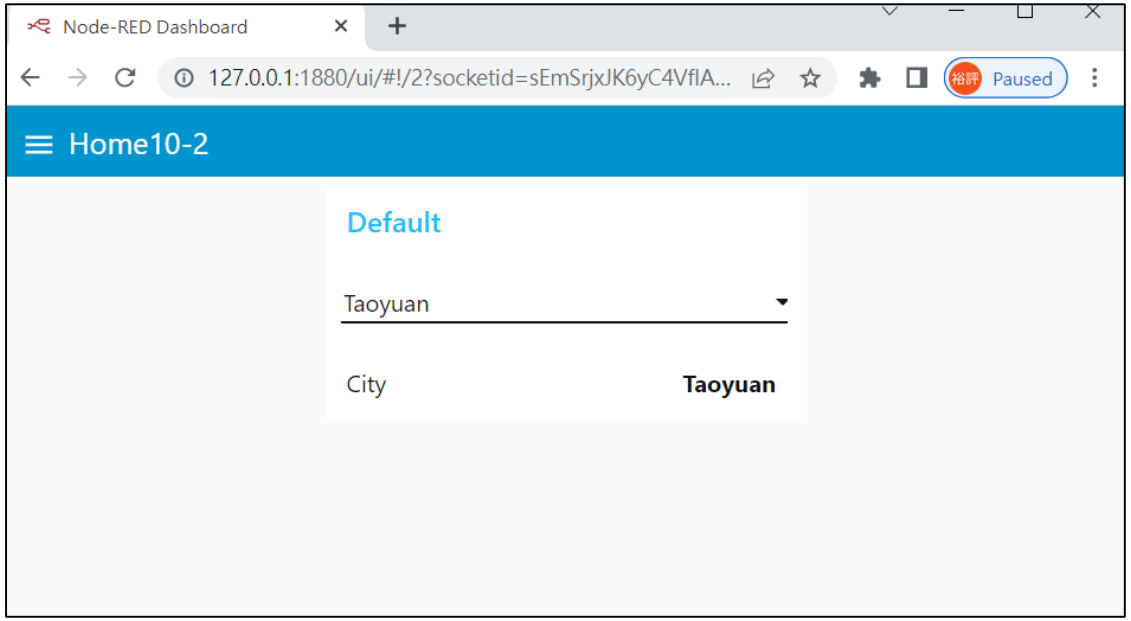

dropdown
debug 3
text
abc

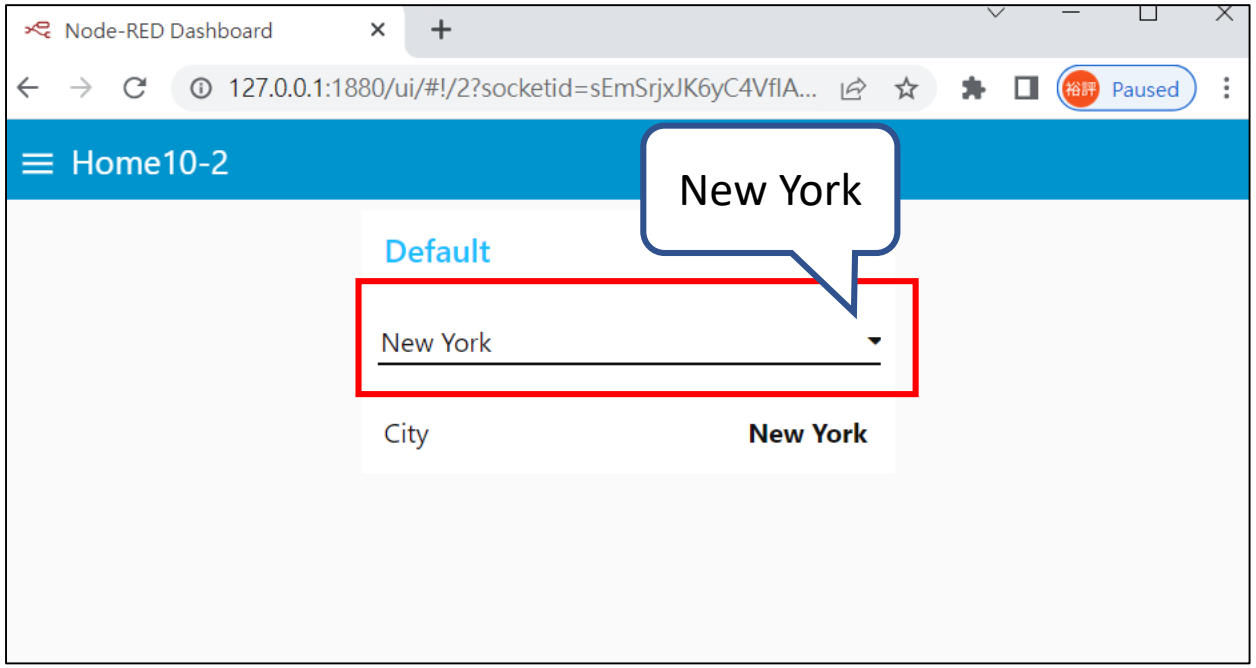

Deploy


3
Edit text node
Delete
Cancel
Done
Properties
Group
[Home10-2] Default
Size
auto
Label
City
Value format
{{msg.payload}}
Layout
label value
label value
label value
label value
label
value
Home10-2
City

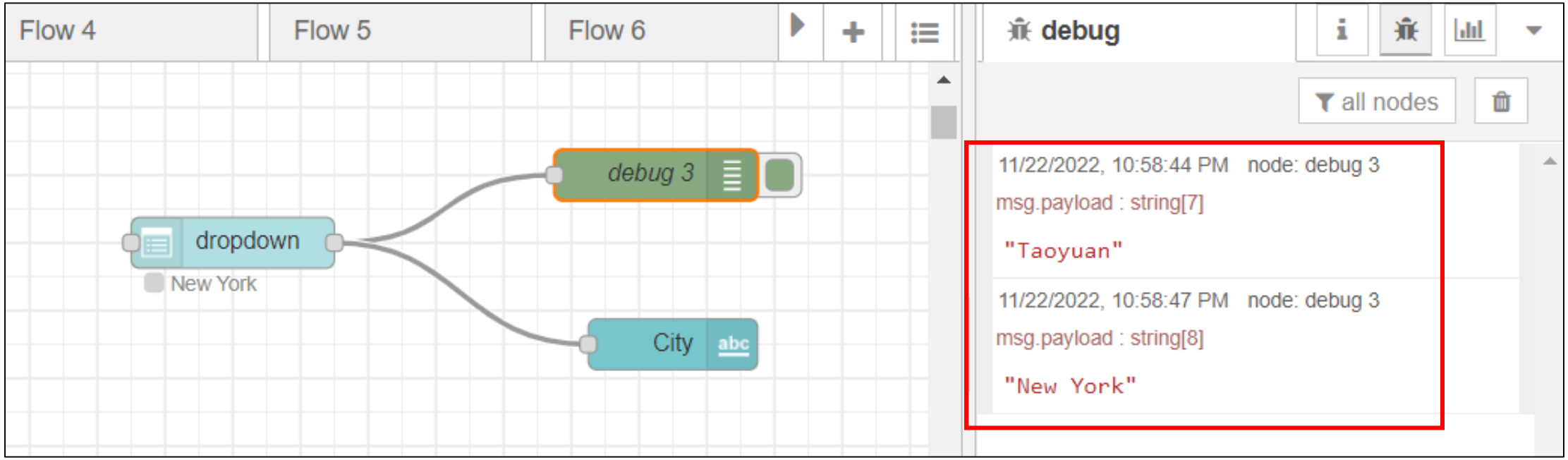

Layout
Site
Theme
Tabs & Links
+tab
+link
Home
Dashboard Demo
Ex9-6
Default
Home10-2
Default
dropdown
City


Node-RED Dashboard
127.0.0.1:1880/ui/#!/2?socketid=sEmSrjxJK6yC4VflA...
Home10-2
Default
Singapur
London
Shanghai
Taipei
Taichung
Taoyuan
New York
Taoyuan


Node-RED Dashboard
127.0.0.1:1880/ui/#!/2?socketid=sEmSrjxJK6yC4VflA...
Paused
Home10-2
Default
Taoyuan
City
Taoyuan

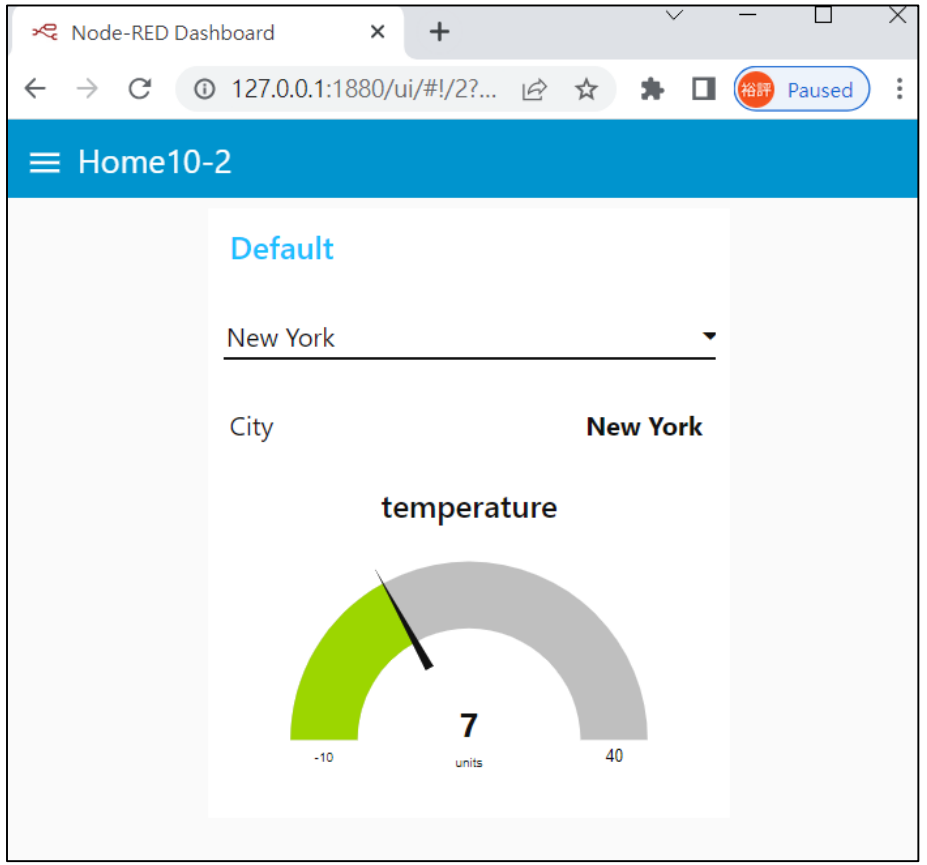

Node-RED Dashboard
127.0.0.1:1880/ui/#!/2?socketid=sEmSrjxJK6yC4VflA...
Paused
Home10-2
New York
Default
New York
City
New York

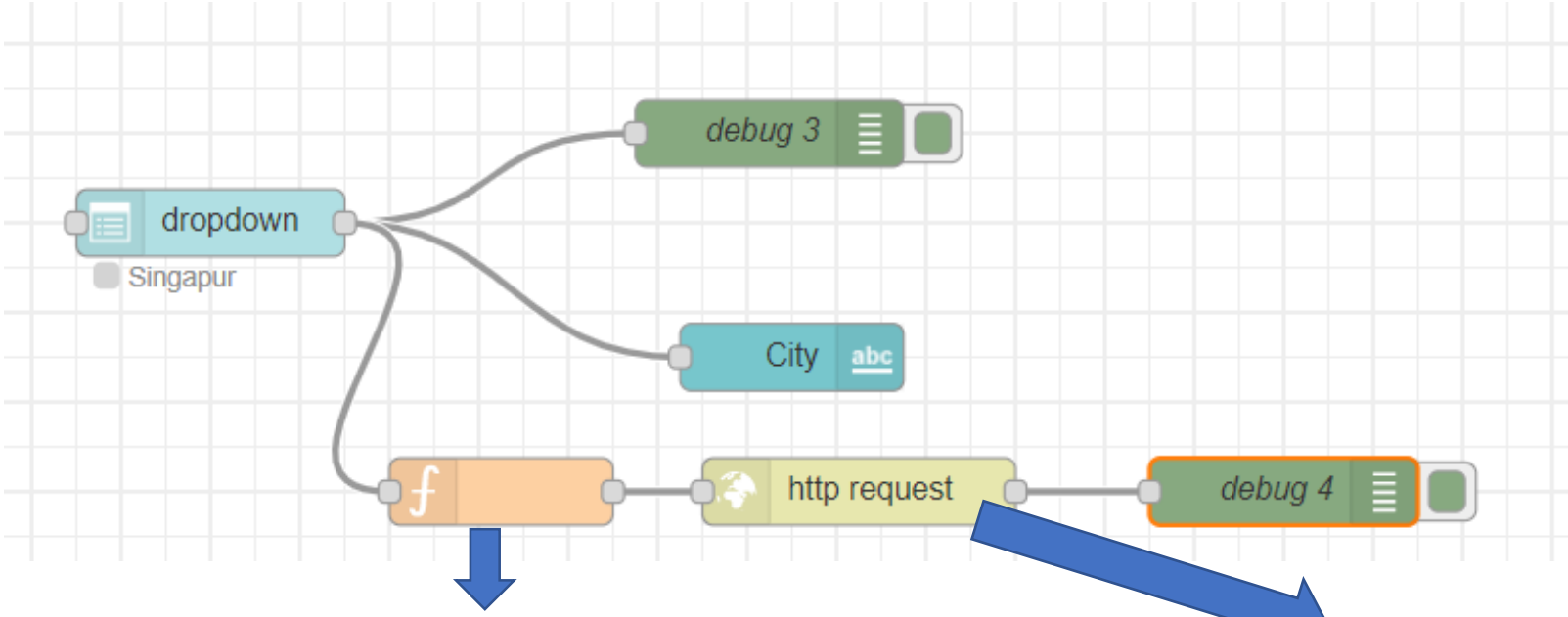
debug


Flow 4
Flow 5
Flow 6
debug 3
dropdown
New York
City
abc
debug
all nodes
11/22/2022, 10:58:44 PM node: debug 3
msg.payload : string[7]
"Taoyuan"
11/22/2022, 10:58:47 PM node: debug 3
msg.payload : string[8]
"New York"

Exercise 10-3

- Show the weather of the selected city on the dashboard.





Edit function node

Delete | Cancel | Done

Properties

Name: Name

Function

```
1  var name=msg.payload;
2
3  msg.url="http://api.weatherstack.com/current?access_key=[redacted]4e&query="+name;
4  return msg;
```

+name

Edit http request node

Delete | Cancel | Done

Properties

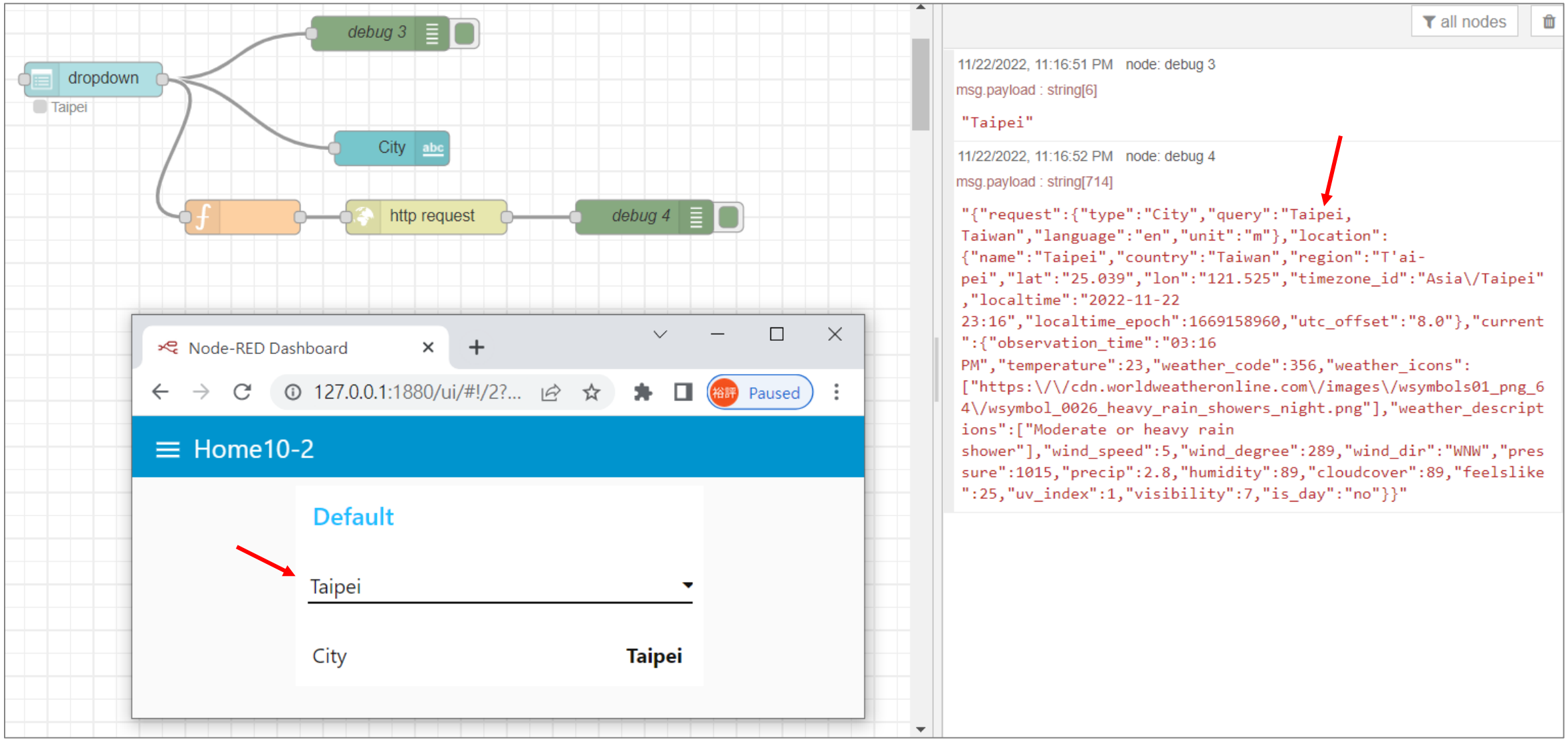
Method: GET

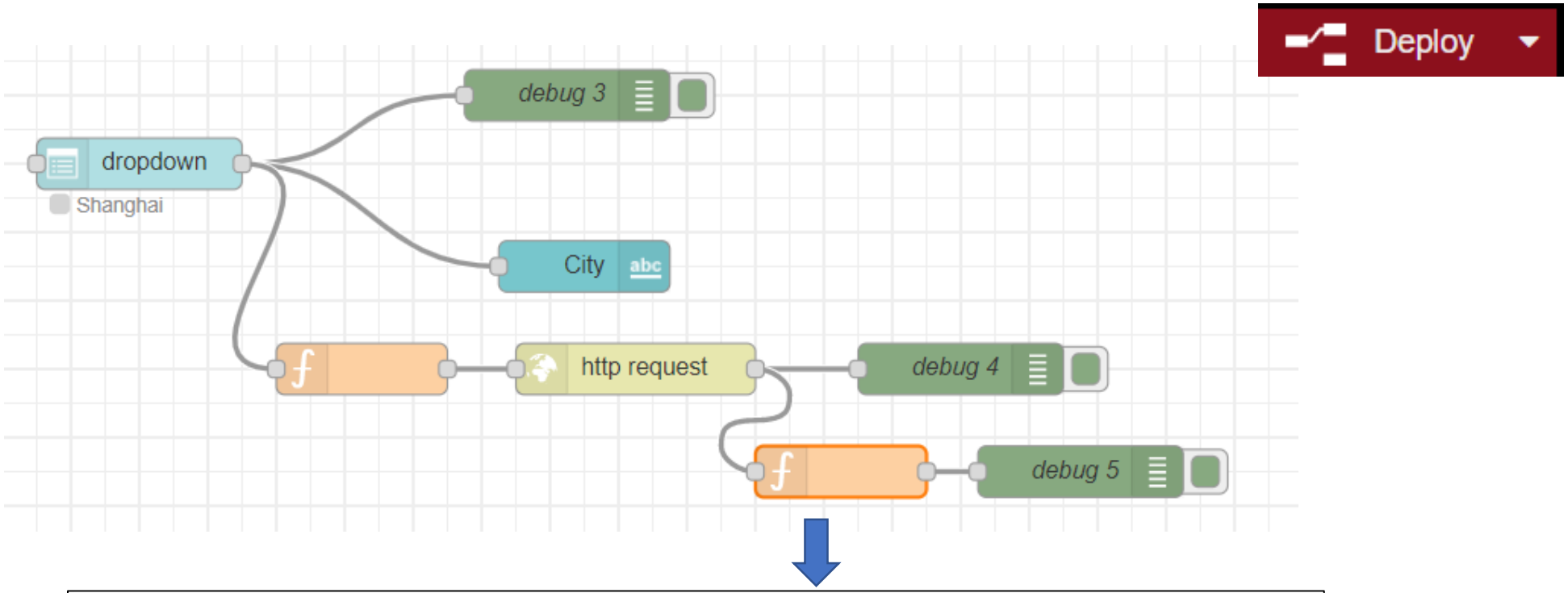
URL: http://

Append msg.payload as query string parameters

Enable secure (SSL/TLS) connection

Use authentication

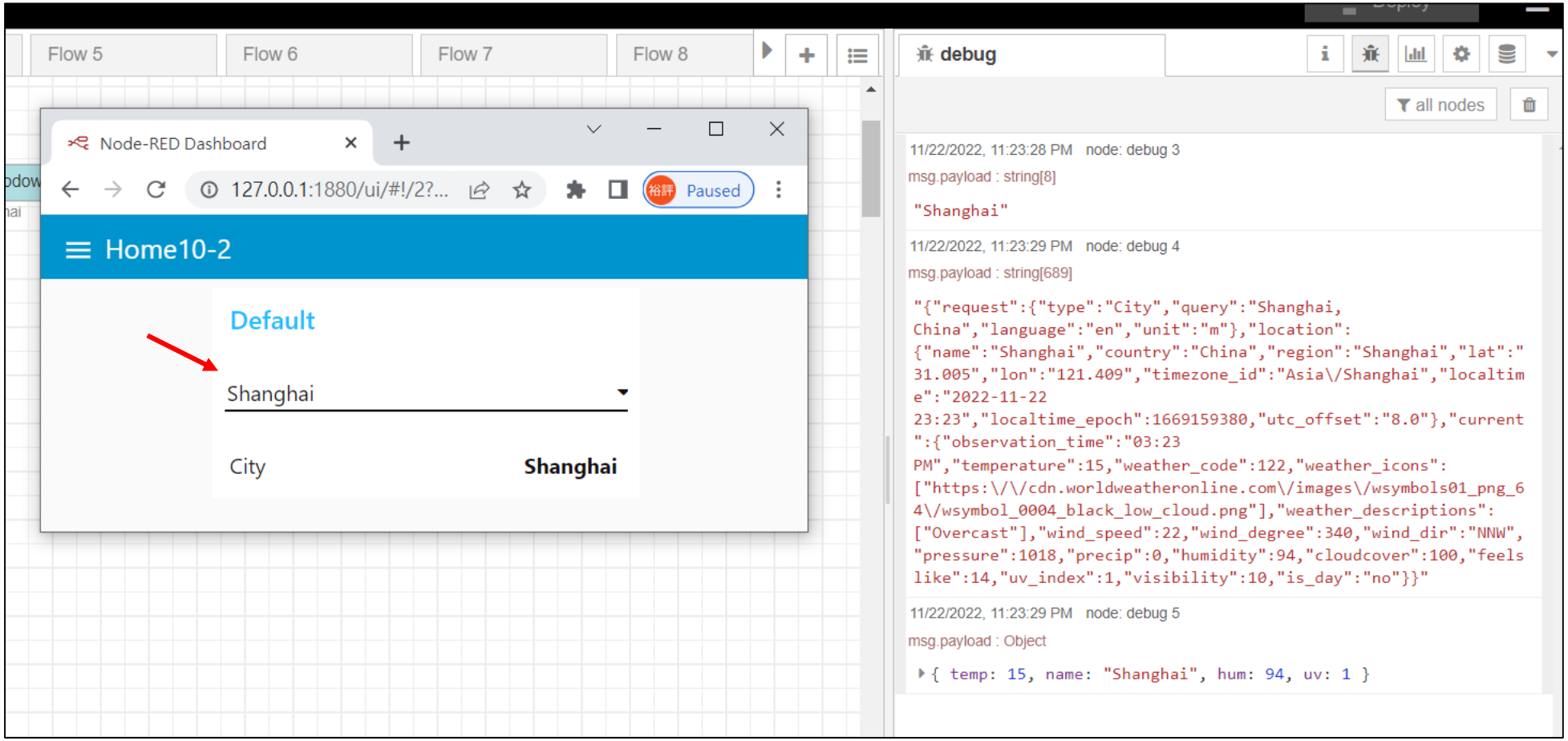

debug 3
dropdown
Taipei
City
abc
http request
debug 4
Node-RED Dashboard
127.0.0.1:1880/ui/#!/2?...
Paused
Home10-2
Default
Taipei
City
Taipei
all nodes
11/22/2022, 11:16:51 PM node: debug 3
msg.payload : string[6]
"Taipei"
11/22/2022, 11:16:52 PM node: debug 4
msg.payload : string[714]
"{"request":{"type":"City","query":"Taipei,
Taiwan","language":"en","unit":"m"},"location":
{"name":"Taipei","country":"Taiwan","region":"T'ai-
pei","lat":"25.039","lon":"121.525","timezone_id":"Asia\/Taipei"
,"localtime":"2022-11-22
23:16","localtime_epoch":1669158960,"utc_offset":"8.0"},"current
":{"observation_time":"03:16
PM","temperature":23,"weather_code":356,"weather_icons":
["https:\/\/cdn.worldweatheronline.com\/images\/wsymbols01_png_6
4\/wsymbol_0026_heavy_rain_showers_night.png"],"weather_descript
ions":["Moderate or heavy rain
shower"],"wind_speed":5,"wind_degree":289,"wind_dir":"WNW","pres
sure":1015,"precip":2.8,"humidity":89,"cloudcover":89,"feelslike
":25,"uv_index":1,"visibility":7,"is_day":"no"}}"



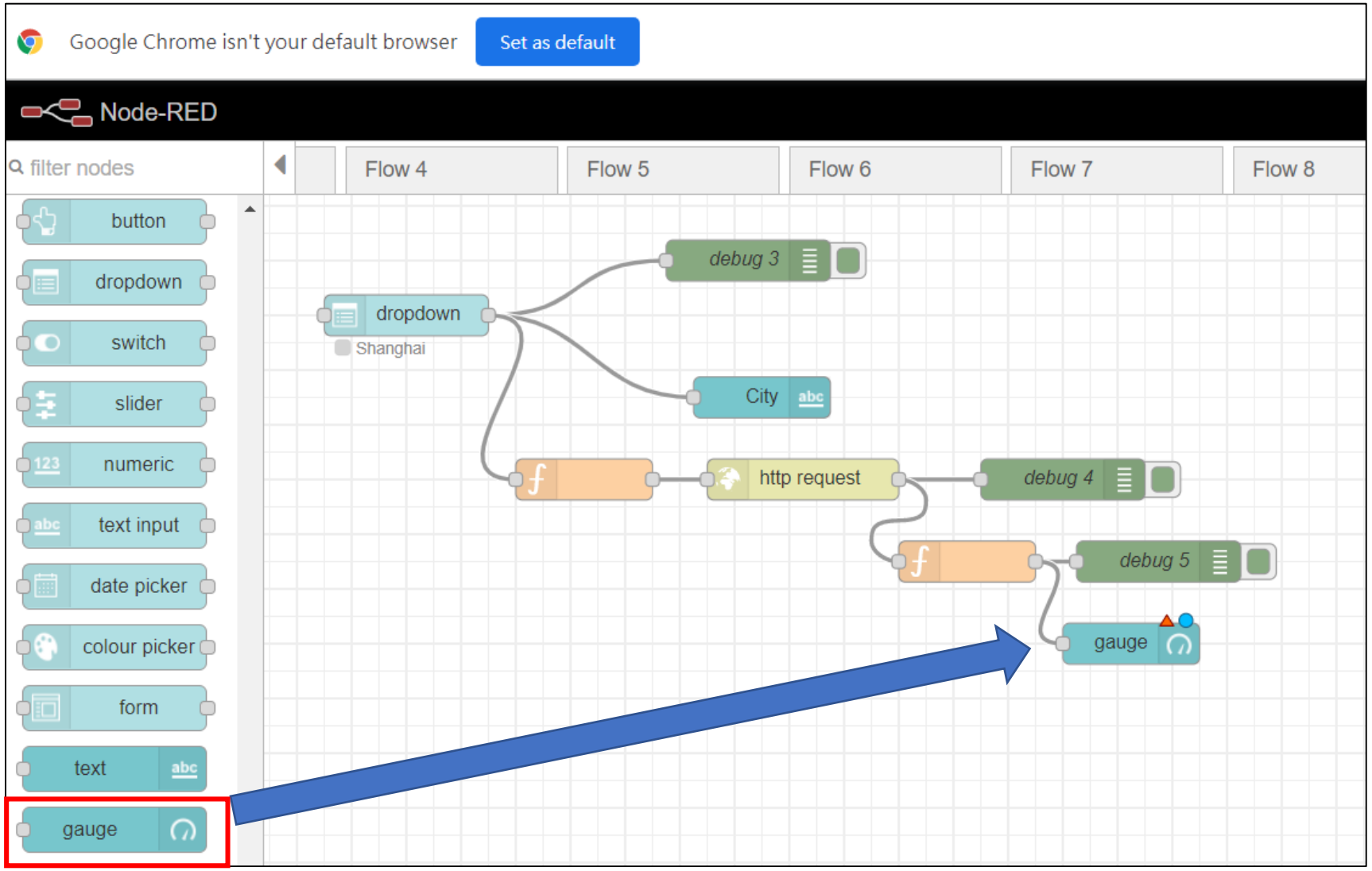
Function

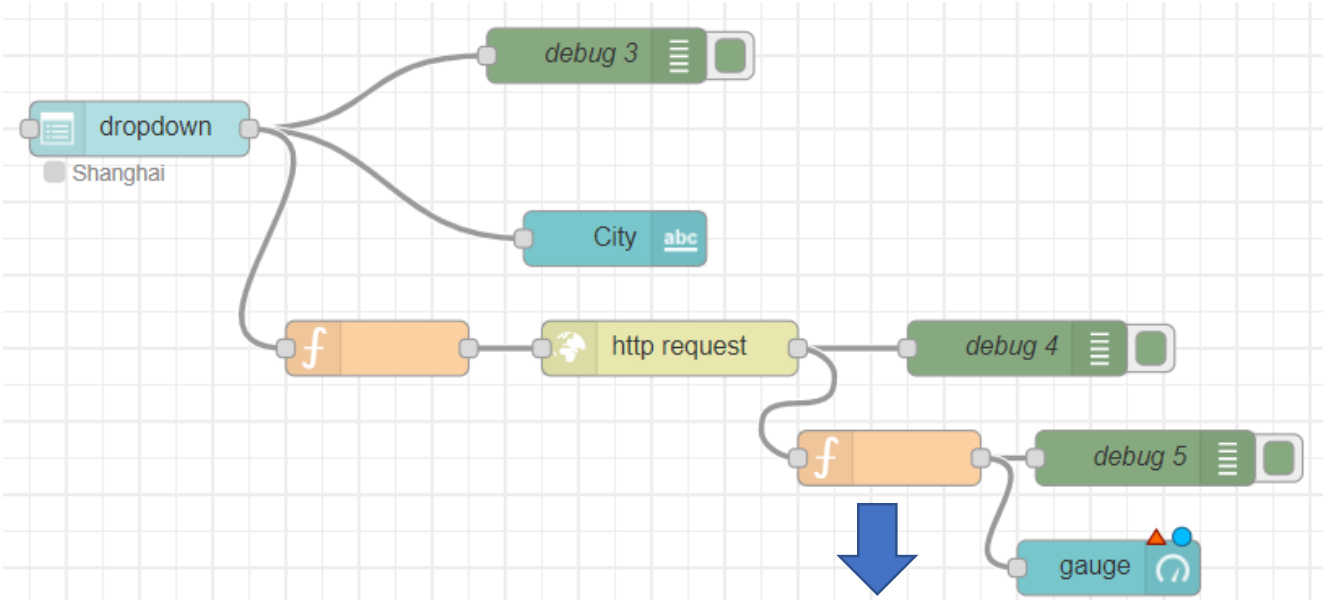
```
1  var result = JSON.parse(msg.payload);
2  var temp = result.current.temperature;
3  var name = result.location.name;
4  var hum = result.current.humidity;
5  var uv = result.current.uv_index;
6  msg.payload = { temp: temp, name: name, hum: hum, uv: uv};
7  return msg;
```

debug


Flow 5
Flow 6
Flow 7
Flow 8
Node-RED Dashboard
127.0.0.1:1880/ui/#!/2?...
Paused
Home10-2
Default
Shanghai
City
Shanghai
debug
all nodes
11/22/2022, 11:23:28 PM node: debug 3
msg.payload : string[8]
"Shanghai"
11/22/2022, 11:23:29 PM node: debug 4
msg.payload : string[689]
"{"request":{"type":"City","query":"Shanghai, China","language":"en","unit":"m"},"location":{"name":"Shanghai","country":"China","region":"Shanghai","lat":"31.005","lon":"121.409","timezone_id":"Asia\/Shanghai","localtime":"2022-11-22 23:23","localtime_epoch":1669159380,"utc_offset":"8.0"},"current":{"observation_time":"03:23 PM","temperature":15,"weather_code":122,"weather_icons":["https:\/\/cdn.worldweatheronline.com\/images\/wsymbols01_png_64\/wsymbol_0004_black_low_cloud.png"],"weather_descriptions":["Overcast"],"wind_speed":22,"wind_degree":340,"wind_dir":"NNW","pressure":1018,"precip":0,"humidity":94,"cloudcover":100,"feelslike":14,"uv_index":1,"visibility":10,"is_day":"no"}}"
11/22/2022, 11:23:29 PM node: debug 5
msg.payload : Object
{ temp: 15, name: "Shanghai", hum: 94, uv: 1 }

gauge


Google Chrome isn't your default browser
Set as default
Node-RED
filter nodes
button
dropdown
switch
slider
numeric
text input
date picker
colour picker
form
text
gauge
Flow 4
Flow 5
Flow 6
Flow 7
Flow 8
debug 3
dropdown
Shanghai
City
http request
debug 4
debug 5
gauge



Deploy

Function

```
1  var result = JSON.parse(msg.payload);
2  var temp = result.current.temperature;
3  var name = result.location.name;
4  var hum = result.current.humidity;
5  var uv = result.current.uv_index;
6  //msg.payload = { temp: temp, name: name, h
7  msg.payload = temp;
8  return msg;
```

Edit gauge node

Delete | Cancel | Done

Properties

Group: [Home10-2] Default

Size: auto

Type: Gauge

Label: temperature

Value format: {{value}}

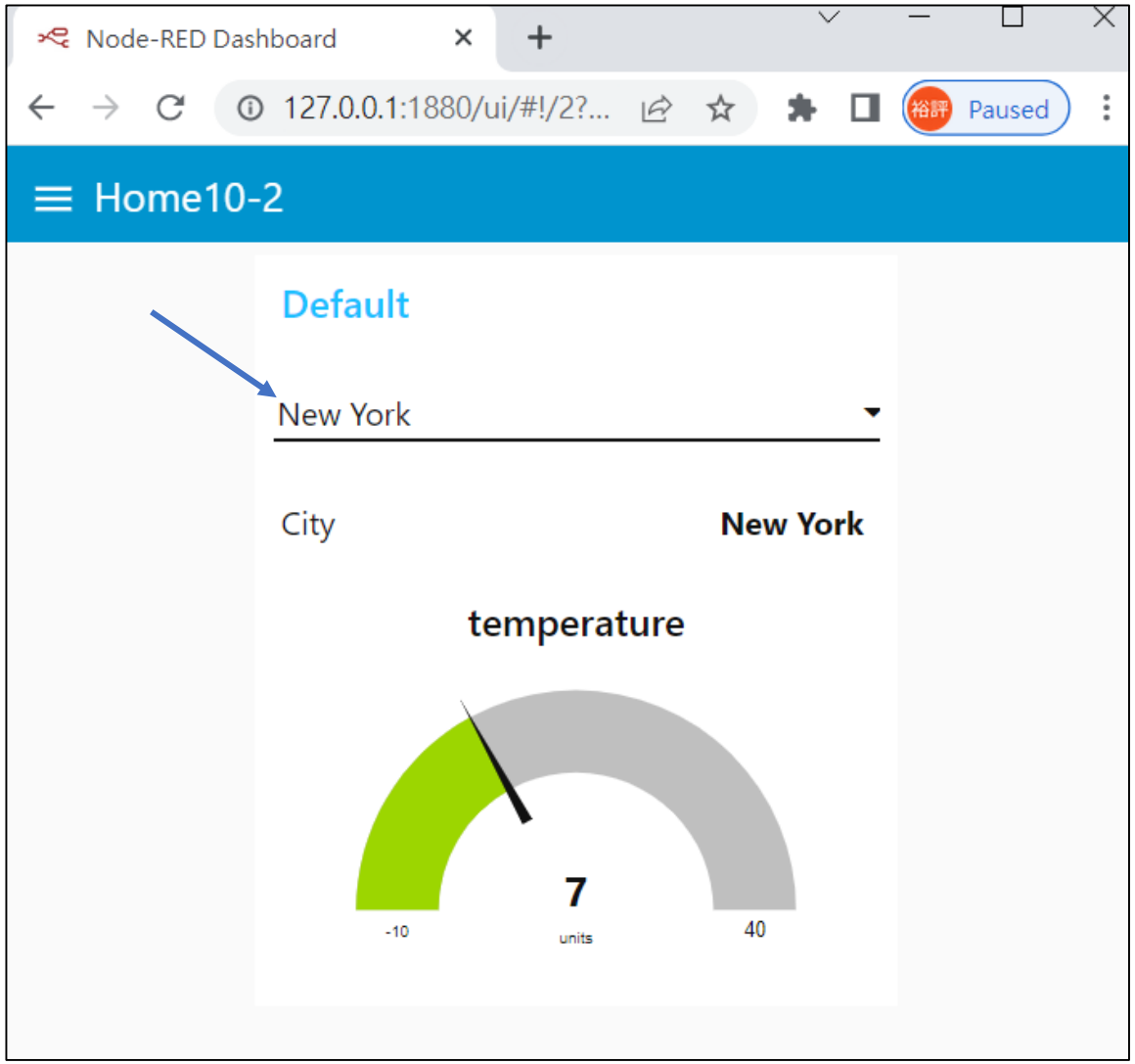
Units: units

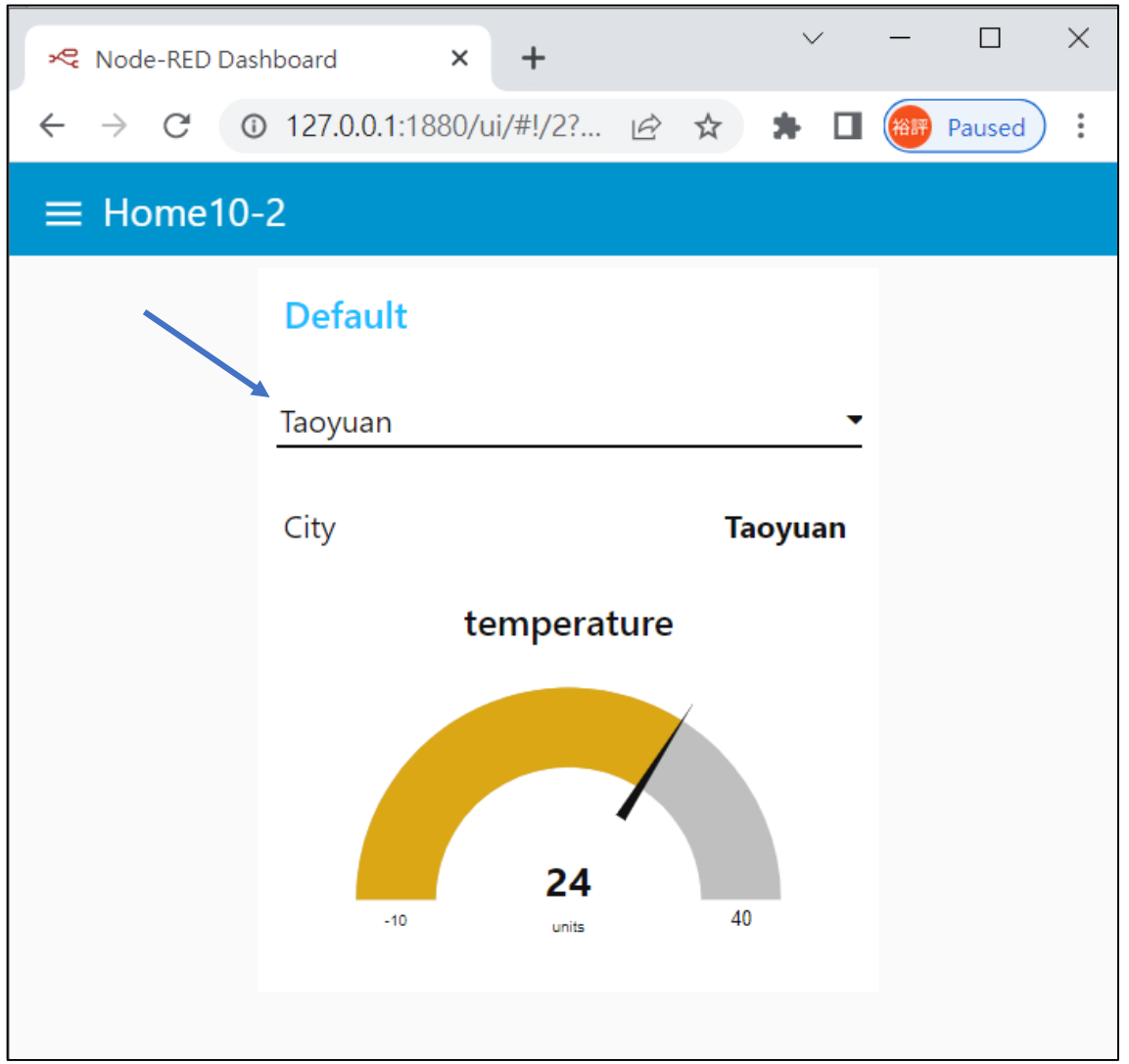
Range: min -10 max 40

temperature

-10

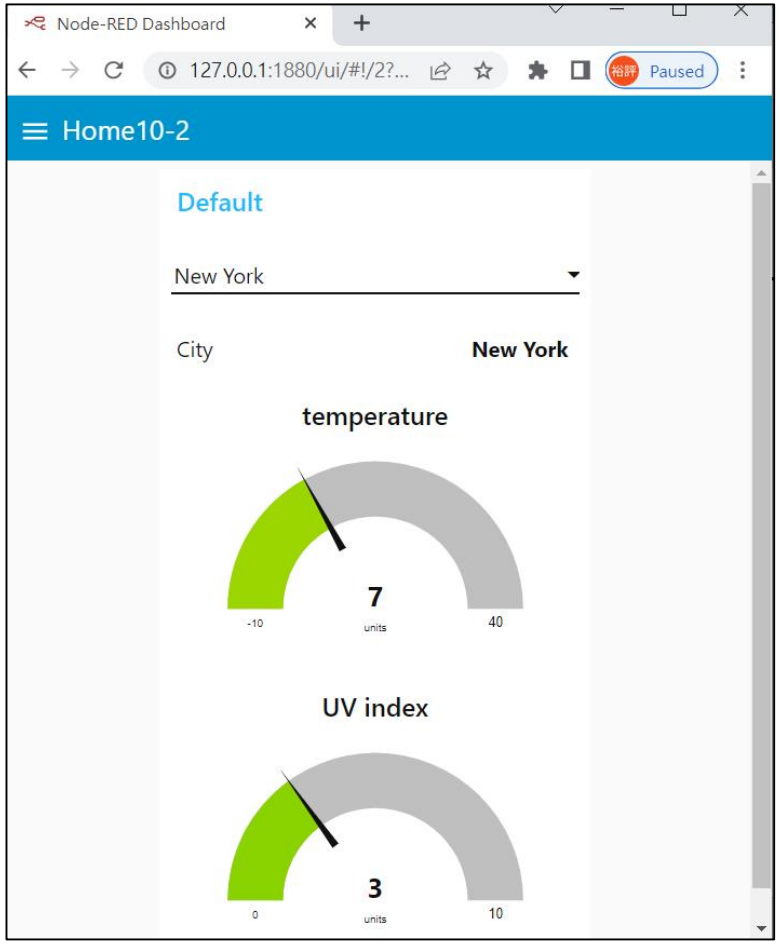
40


Node-RED Dashboard
127.0.0.1:1880/ui/#!/2?...
Paused
Home10-2
Default
New York
City
New York
temperature
7
-10
units
40


Node-RED Dashboard
127.0.0.1:1880/ui/#!/2?...
Paused
Home10-2
Default
Taoyuan
City
Taoyuan
temperature
24
-10
units
40

Exercise 10-4

- Show the UV index and temperature of the selected city on the dashboard.



Exercise 10-5

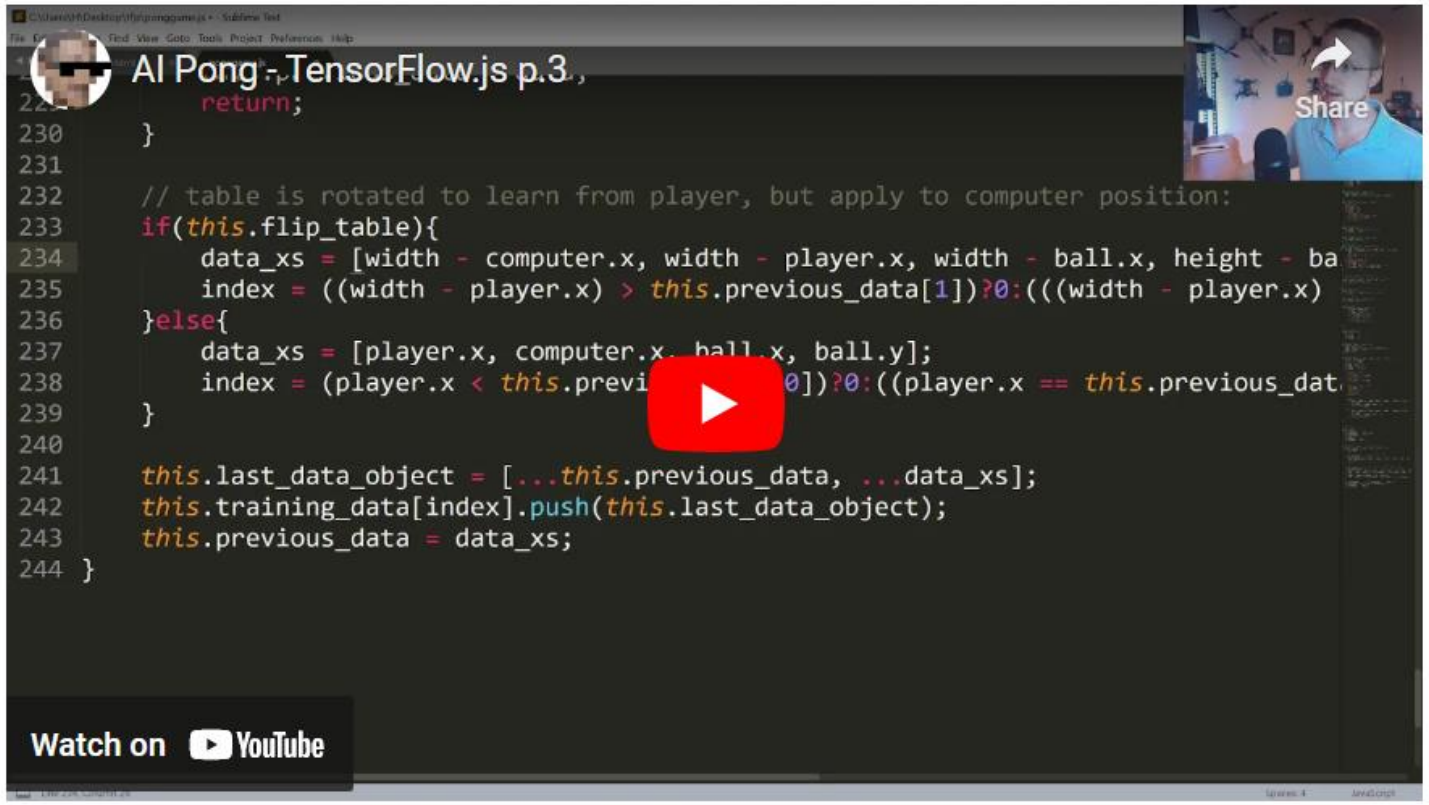
- Try another weather datasets to establish a weather station
- For example:
- https://opendata.cwb.gov.tw/dataset/climate?page=1

Hamdo-利用即時骨架辨識及感測器輔助分析劍道訓練與運動管理系統之開發

https://www.youtube.com/watch?v=VtJEemY03VA

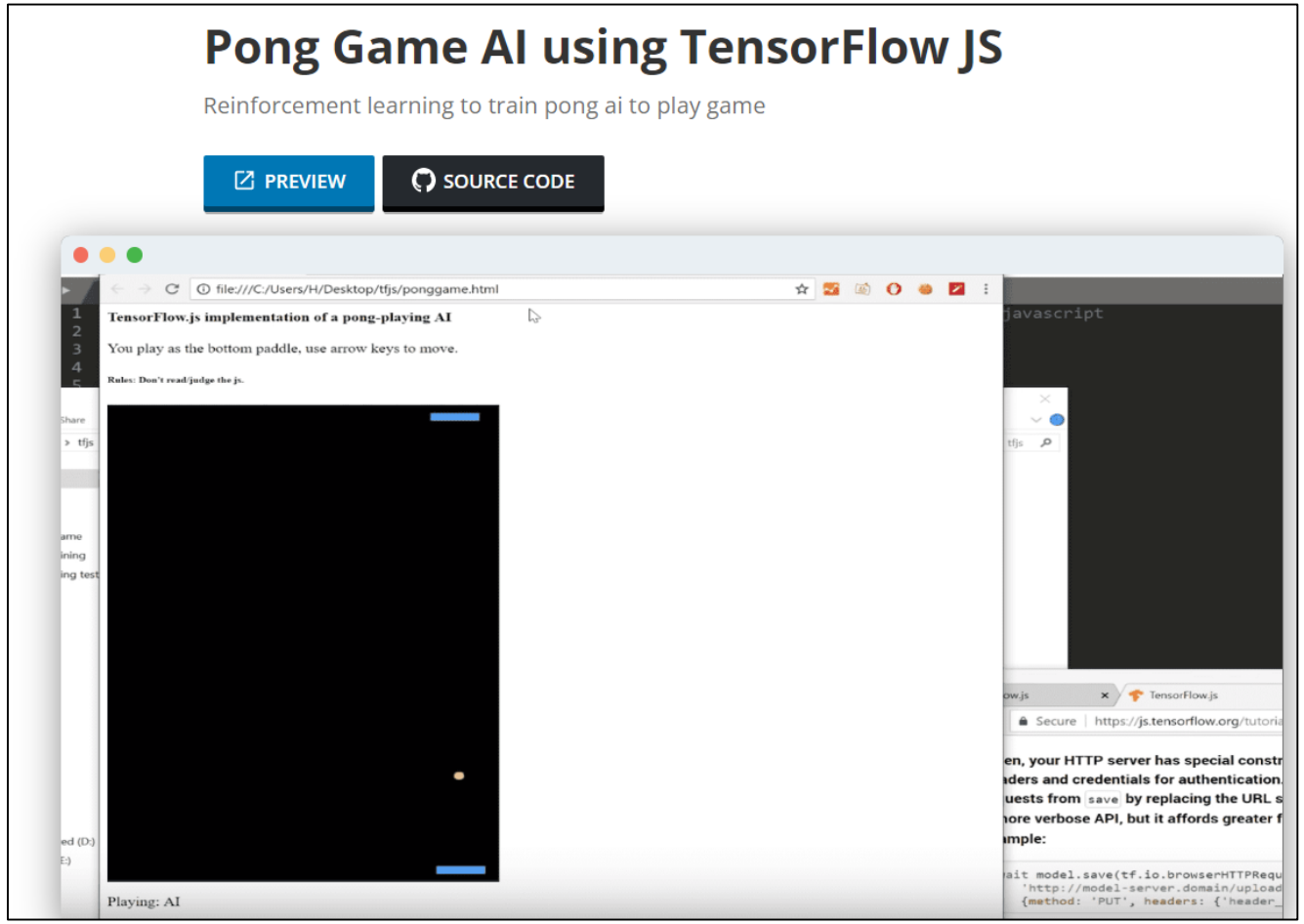
AI Game

- https://pythonprogramming.net/pong-ai-tensorflowjs-tutorial/



Pong Game AI using TensorFlow JS

- https://abhimanyuaryan.com/portfolio/reinforcement-learning



Tensorflow.js