智慧整合感控系統概論
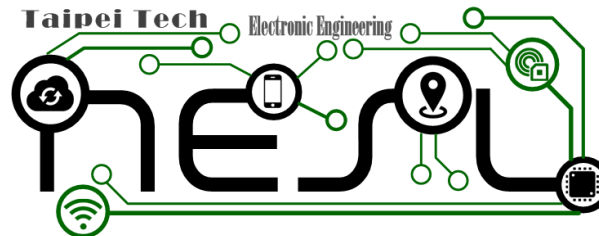# Introduction to Cyber-Physical Systems

LAB：OM2M + NodeRED

國立臺北科技大學電子工程系
授課教師：李昭賢 副教授
電子郵件：chlee@ntut.edu.tw
校內分機：2288

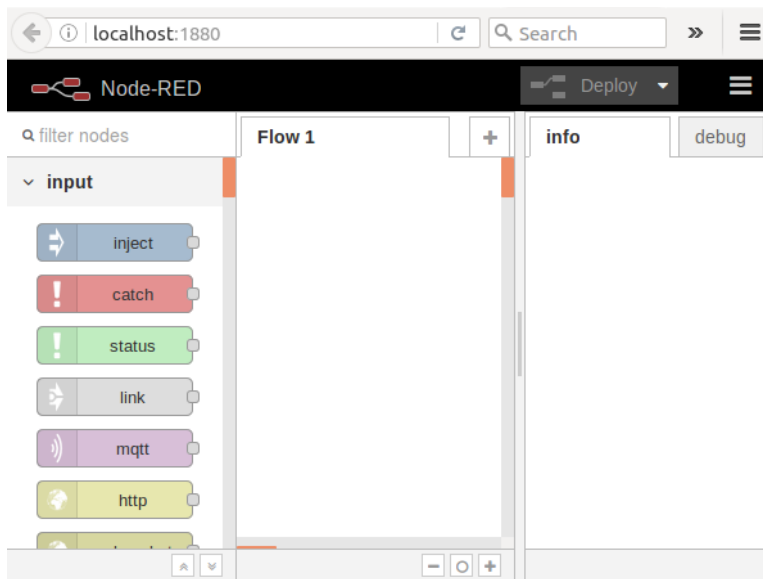| 1	利用NodeRED存取OM2M

NodeRED與OM2M

❖ 本次上課的VM內已提供NodeRED中OM2M的套件

- Open a new terminal and input the command:
  - $ sudo node-red -vv
- Use any browser to open http://localhost:1880
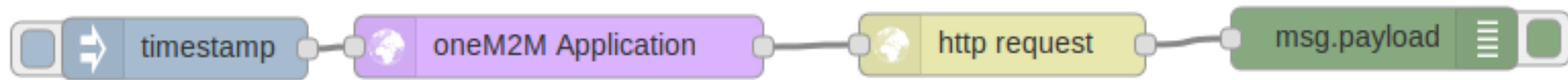


```
Welcome to Node-RED
===================

28 Jun 15:34:34 - [info] Node-RED version: v0.14.3
28 Jun 15:34:34 - [info] Node.js  version: v4.2.6
28 Jun 15:34:34 - [info] Linux 4.4.0-21-generic x64 LE
28 Jun 15:34:34 - [info] Loading palette nodes
28 Jun 15:34:36 - [warn] ------------------------------------------------
28 Jun 15:34:36 - [warn] [rpi-gpio] Info : Ignoring Raspberry Pi specific node
28 Jun 15:34:36 - [warn] ------------------------------------------------
28 Jun 15:34:36 - [info] Settings file  : /home/iotclass/.node-red/settings.js
28 Jun 15:34:36 - [info] User directory : /home/iotclass/.node-red
28 Jun 15:34:36 - [info] Flows file     : /home/iotclass/.node-red/flows_iotclas
s.json
28 Jun 15:34:36 - [info] Creating new flow file
28 Jun 15:34:36 - [info] Starting flows
28 Jun 15:34:36 - [info] Started flows
28 Jun 15:34:36 - [info] Server now running at http://127.0.0.1:1880/
```

❖ NodeRED

- ■ "inject" from the input library
- ■ "oM2M ONE Application" from the function library
- ■ "http request" from the function library
- ■ "debug" from the output library
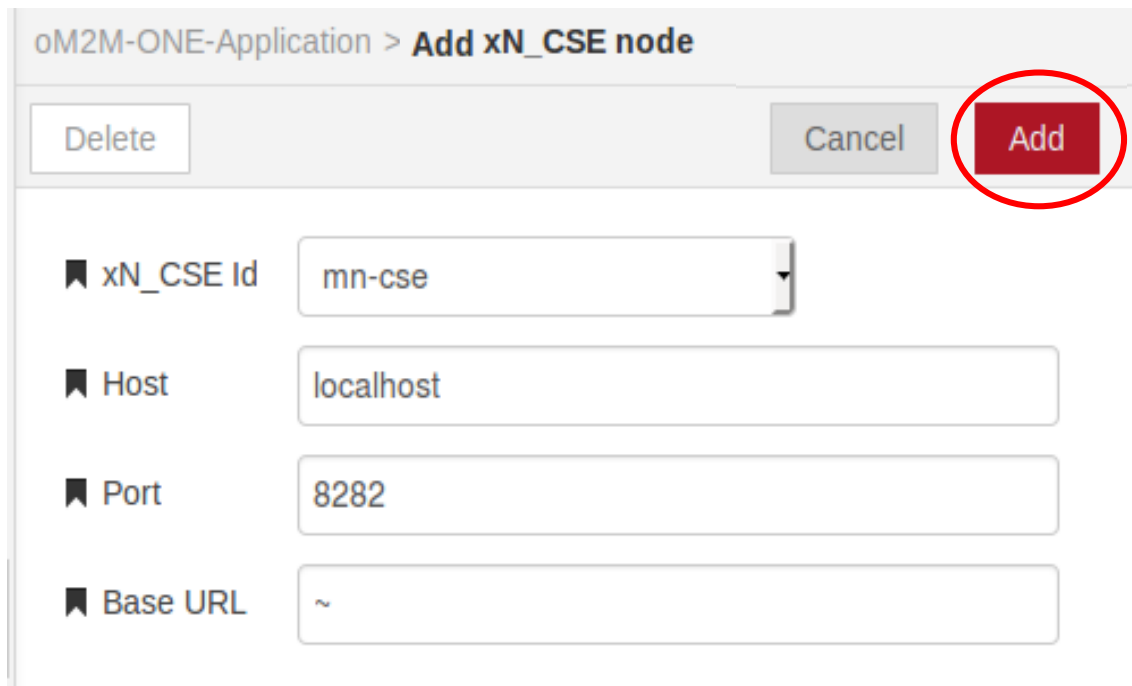
❖ NodeRED

- ■ edit xN_CSE attribute contents

❖ NodeRED

- ■ "Add new xN_CSE node" according to the information shown in the picture below

❖ NodeRED

- Choose "http://localhost:8282" from the xN_CSE list (You just created this item in the previous step).
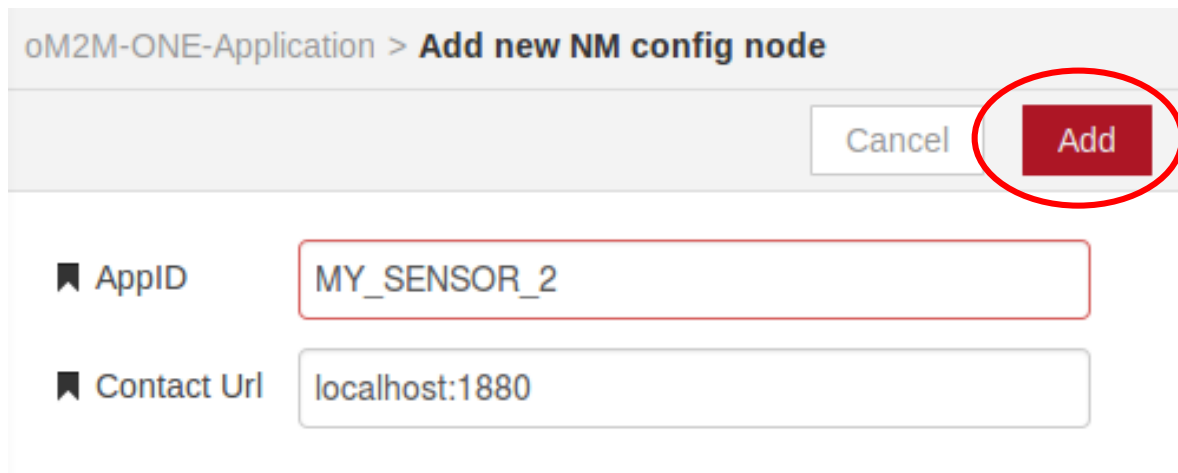
❖ NodeRED

- ▪ "Add new NM config node" with the information shown in the picture below.

❖ NodeRED

- ▪ Choose "MY_SENSOR_2" from the NM list (You just created this item in the previous step).

❖ NodeRED

- **Choose "POST" from the Method list.**
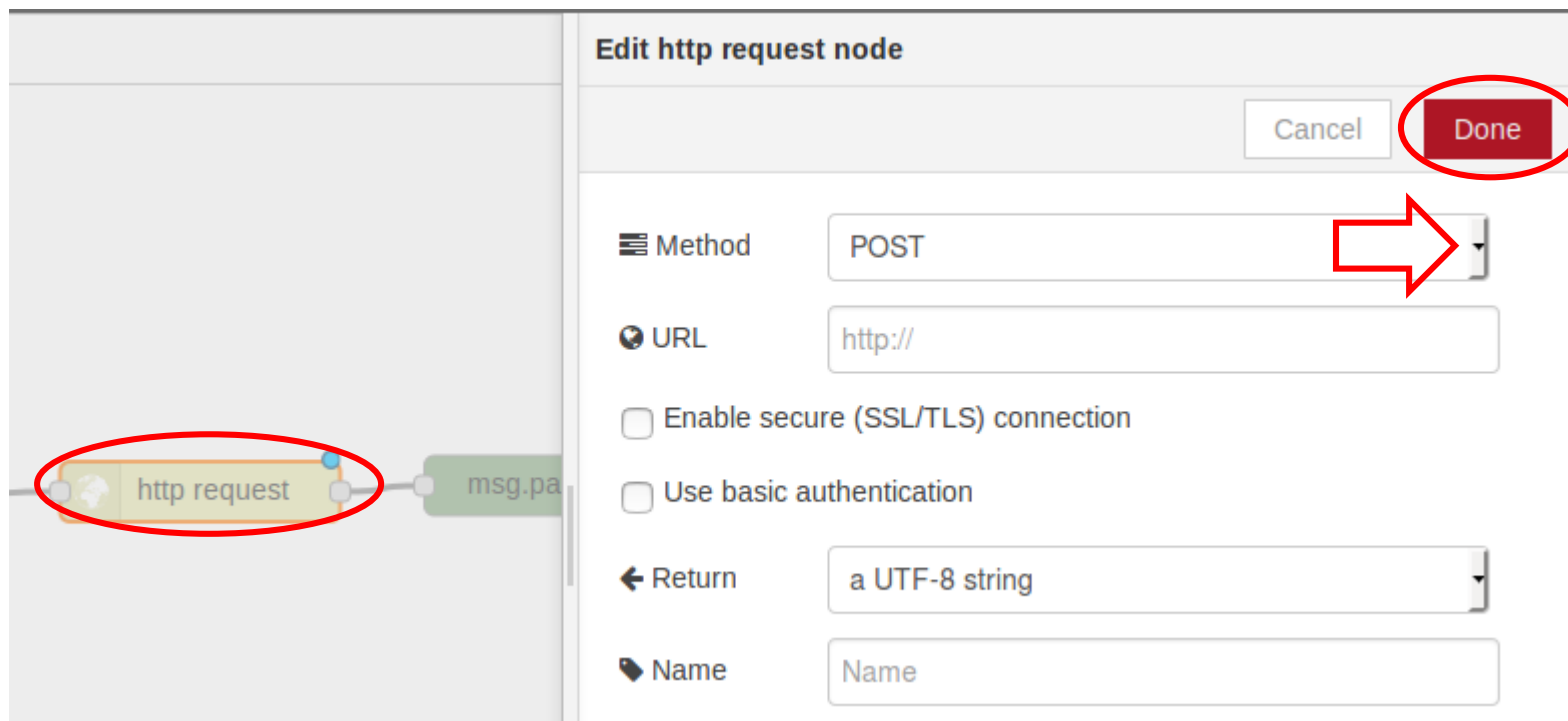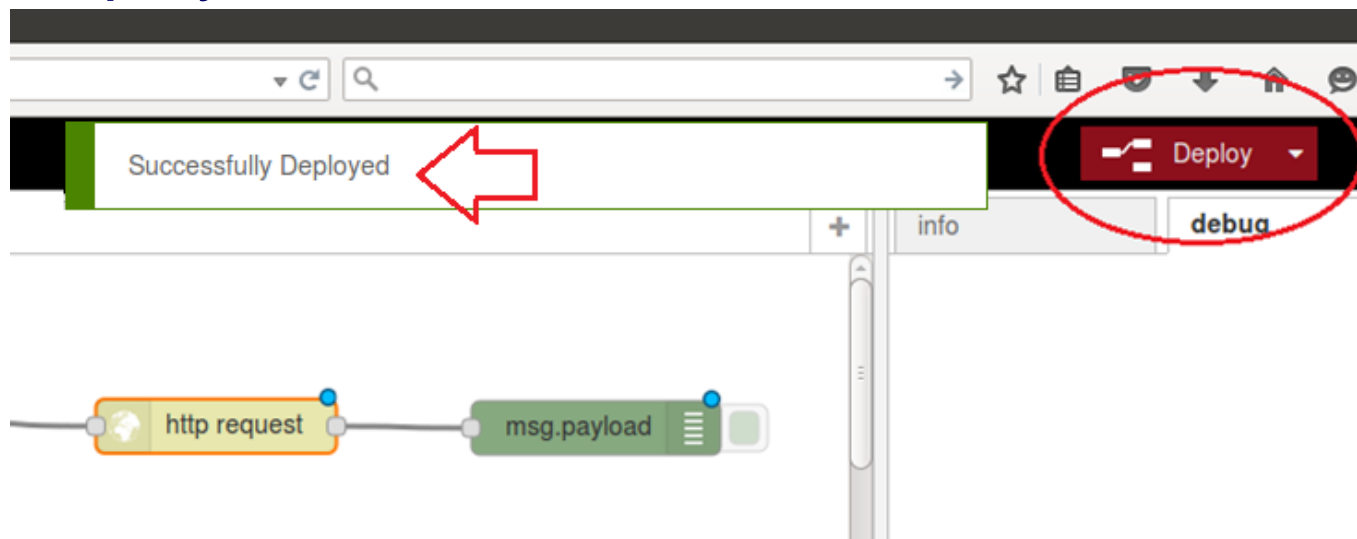
❖ NodeRED

■ Deploy & Execute
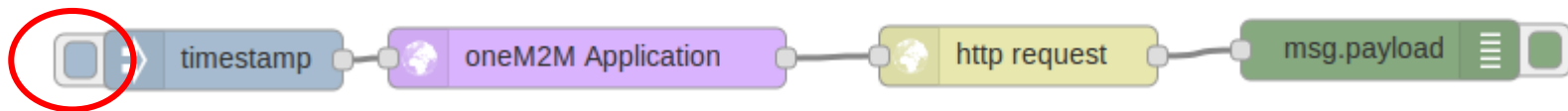
❖ Open OM2M Resource Tree Navigation Tool for MN-CSE

OM2M CSE Resource Tree

http://127.0.0.1:8282/~/mn-cse

```
─ mn-name
    ├ acp_admin
    ├ acpae-344339622
    ├ acpae-819133182
    ├ MY_SENSOR
    ├ MY_SENSOR_2
    └ in-name
```

Attribute	Value
ty	2
ri	/mn-cse/CAE819133182
pi	/mn-cse
ct	20160630T151701
lt	20160630T151701
lbl	Type/Demo Category/GatewayApplication Location/NCTU
acpi	**AccessControlPolicyIDs** /mn-cse/acp-410481084
et	20170630T151701
api	app-sensor
aei	CAE819133182
rr	false

❖請利用OM2M Resource Tree Visualizer Tool顯示練習一的結果。

❖ NodeRED

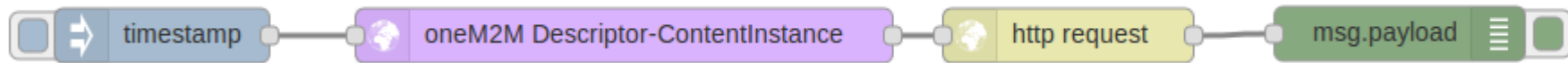- Repeat the steps of the previous exercise to complete these tasks.
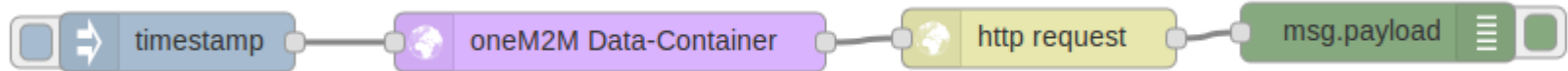 - Descriptor Container
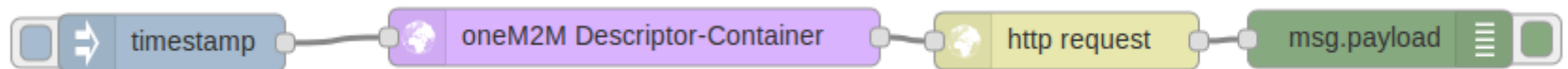
  - Descriptor Content Instance

❖ NodeRED

- Repeat the steps of the previous exercise to complete these tasks.

  - Data Container

  - Data Content Instance

練習二：Create Container & ContentInstance

❖ NodeRED

- ■ Add the details about "My Data"


```
CODE
var data = {
    type:'string',
    unit:'test data',
    data: 'Hello World!'
};
msg.payload =
JSON.stringify(data);
return msg;
```
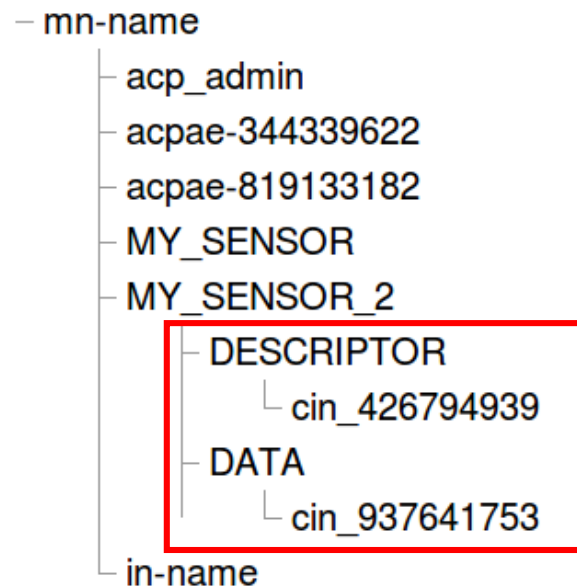
❖ Open OM2M Resource Tree Navigation Tool for MN-CSE

OM2M CSE Resource Tree

http://127.0.0.1:8282/~/mn-cse/cnt-743761537

- mn-name
  - acp_admin
  - acpae-344339622
  - acpae-819133182
  - MY_SENSOR
  - MY_SENSOR_2
    - DESCRIPTOR
      - cin_426794939
    - DATA
      - cin_937641753
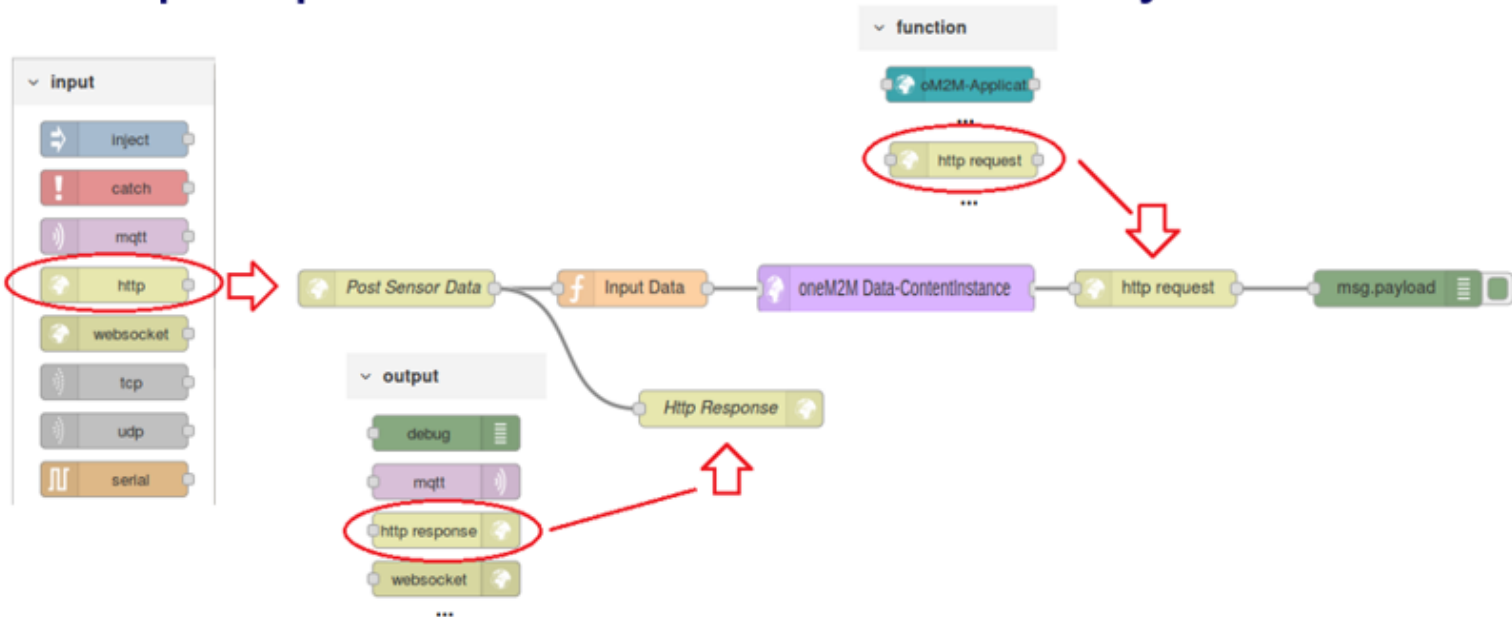 - in-name

❖ 請利用OM2M Resource Tree Visualizer Tool顯示練習二的結果。

❖ PS: 以上練習等同於之前使用Postman練習產生一 MY_SENSOR Application (但比較快速和簡單)

❖ NodeRED

- "http" from the input library
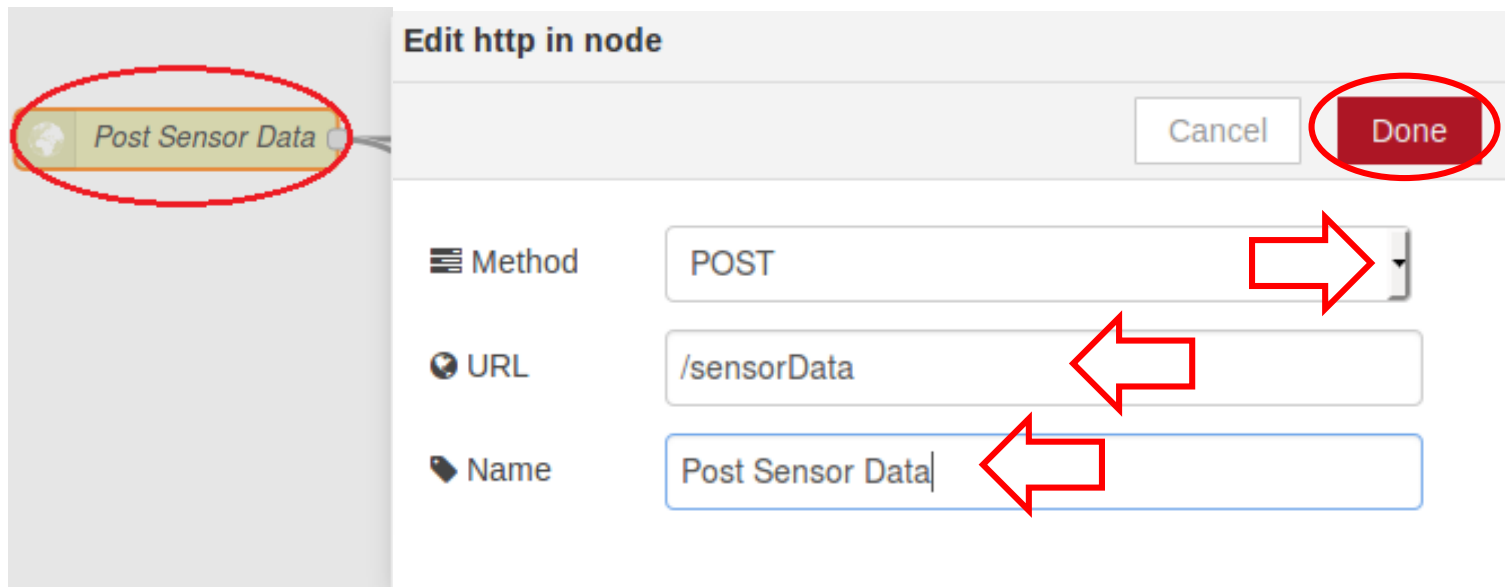- "http response" from the output library
- "http request" from the function library

❖NodeRED

- Fill out the form according to the picture

❖ With this configuration, we are creating a server that listens for HTTP POST requests in the following address:
http://localhost:1880/sensorData.

- Remember that http://localhost:1880 is the address of Node-RED. All other web services that we create are opened under the Node-RED address.

❖ NodeRED

- Fill out the form according to the picture.

CODE
```
var data = {
    type:'string',
    unit:'fake data',
    data: msg.payload.name
};
msg.payload =
JSON.stringify(data);
return msg;
```

❖ In order to test our server, please send an HTTP POST (using Postman) to http://localhost:1880/sensorData.

- You must include a JSON object similar to this example: {"name": "TYPE YOUR NAME HERE"} in the body of the request.

- In Postman, set the data type to "application/json".

- No need to add Authentication Information.

❖ 請利用OM2M Resource Tree Visualizer Tool顯示練習三的結果。

❖ NodeRED

- **Create a IN-AE in the IN-CSE**

```
xN_CSE:
        xN_CSE Id= in-cse,
        Host = localhost,
        port = 8080,
NM: AppID = MY_NETWORK_APPLICATION,
Type = NetworkApplication
Category = demo
Location = Cloud
```

❖ NodeRED

- Create a subscription from the IN-AE into the MN-AE
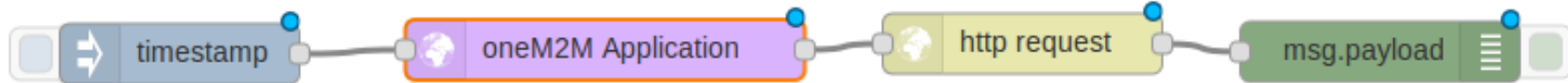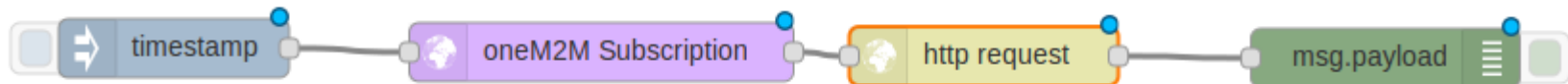
練習四：Subscribe to Data

❖ NodeRED

- Complete the "http request" object according to the picture.

❖ Open OM2M Resource Tree Navigation Tool

OM2M CSE Resource Tree

http://127.0.0.1:8080/~/in-cse/CAE23111059

```
– in-name
    acp_admin
    acpae_23111059
    MY_NETWORK_APPLICATION
    mn-cse
```

OM2M CSE Resource Tree

http://127.0.0.1:8282/~/mn-cse/cnt-743761537
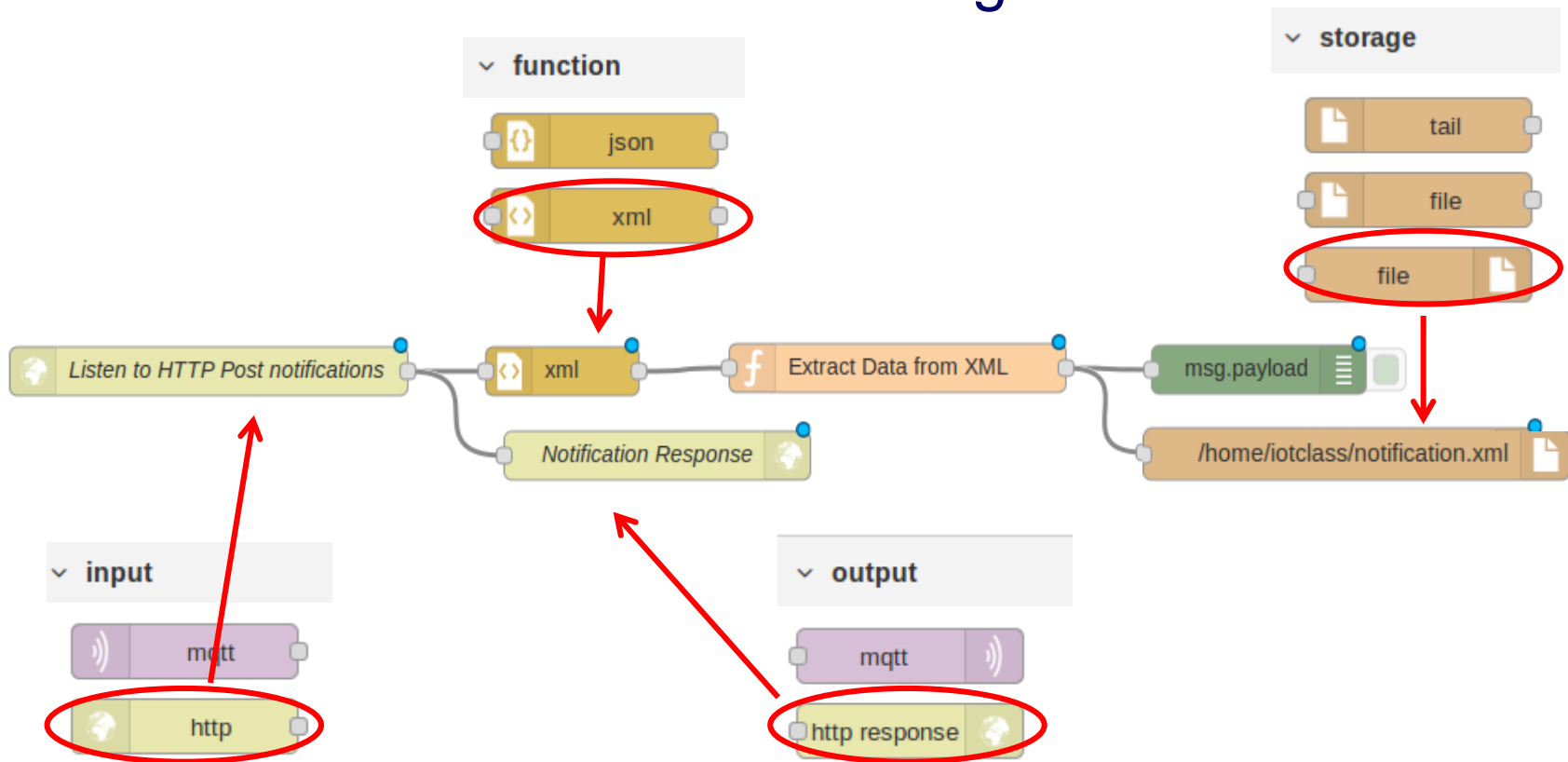
```
– mn-name
    acp_admin
    acpae-344339622
    acpae-819133182
    MY_SENSOR
    MY_SENSOR_2
        DATA
            cin_937641753
            cin_138762453
            SUB_DATA
        DESCRIPTOR
    in-name
```

❖ NodeRED

- **Create a web service listening for notifications**

❖ NodeRED

- Complete its form according to the picture.

❖NodeRED

- Complete its form according to the picture.

Edit function node

Cancel Done

🏷 Name   Extract Data from XML

🔧 Function

```
1  var notification = msg.payload['m2m:sgn'];
2  var nev = notification['nev'][0];
3  var rep = nev['rep'][0];
4  var con = rep['con'][0];
5  msg.payload = con;
6  return msg;
7
```

CODE
```
var notification = msg.payload['m2m:sgn'];
var nev = notification['nev'][0];
var rep = nev['rep'][0];
var con = rep['con'][0];
msg.payload = con;
return msg;
```

⤨ Outputs   1

行動寬頻尖端技
術跨校教學聯盟

❖NodeRED

- Complete its form according to the picture.

❖ At this moment, we have created a web service which is listening to HTTP POST requests in the following address: http://localhost:1880/notification.

  ▪ In order to insert new data to "MY_SENSOR_2" MN-AE make an HTTP POST (using Postman) to the address http://localhost:1880/sensorData including a JSON object similar to {"name": "TYPE YOUR NAME HERE"}.

  ▪ No Authentication Information is needed.

行動寬頻尖端技術跨校教學聯盟

❖ NodeRED

- Extend your Network Application, creating an HTTP GET web service in order to retrieve the content of the XML file which is the result of the previous exercise.


```
Read XML File via HTTP GET    /home/iotclass/notification.xml    <> xml    http
```
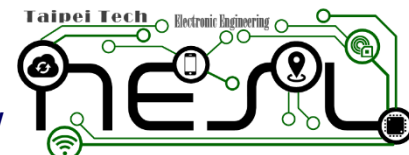
```
localhost:1880/notification
```
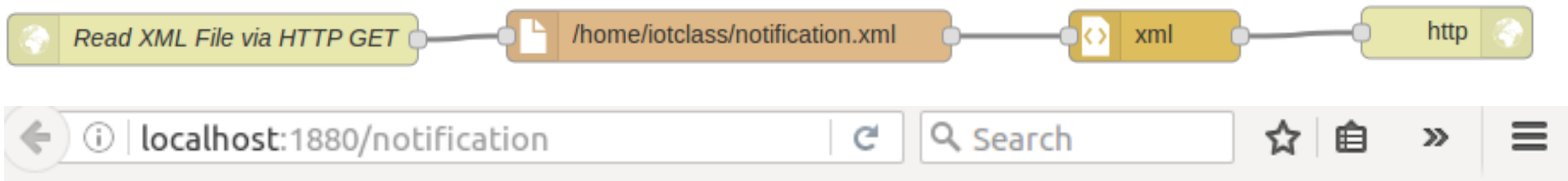
{"obj":{"str":[{"$":{"name":"appId","val":"MY_SENSOR_2"}},
{"$":{"name":"category","val":"GatewayApplication"}}],"int":[{"$":{"name":"data","val":"Check"}},
{"$":{"name":"unit","val":"fake data"}}]}}

❖ 請使用網頁顯示練習四的結果。
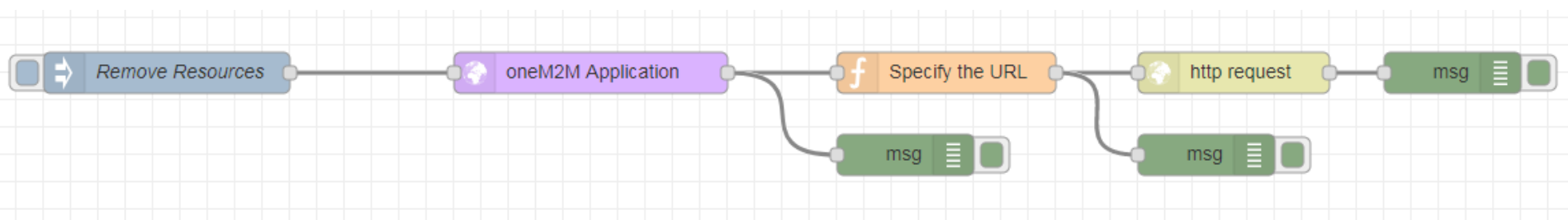
❖ 刪除特定資源 - 指定URL
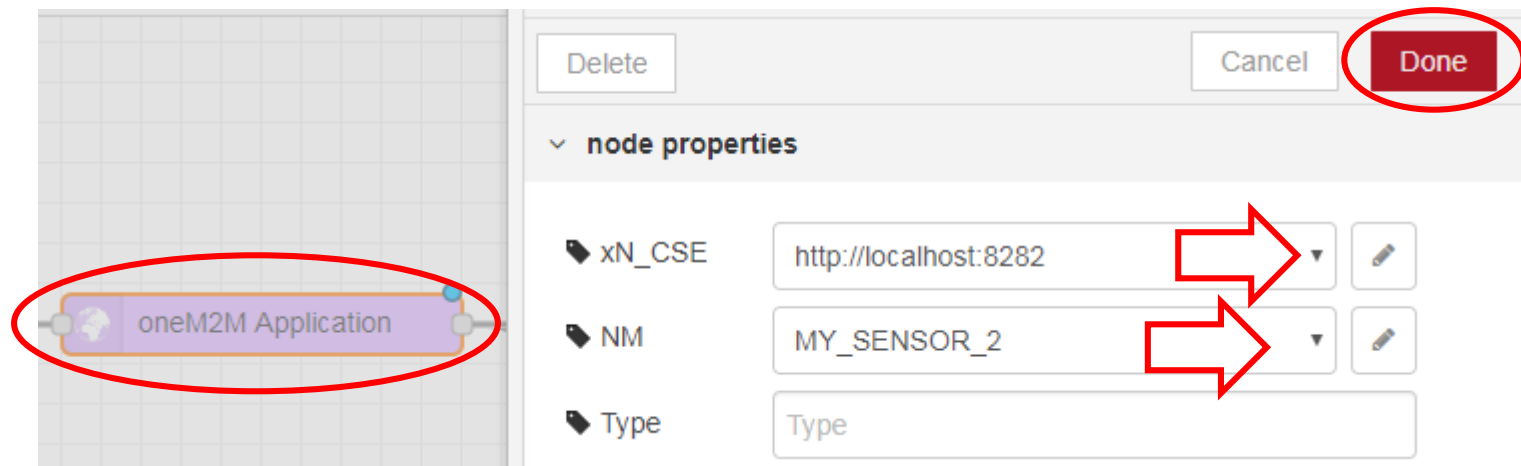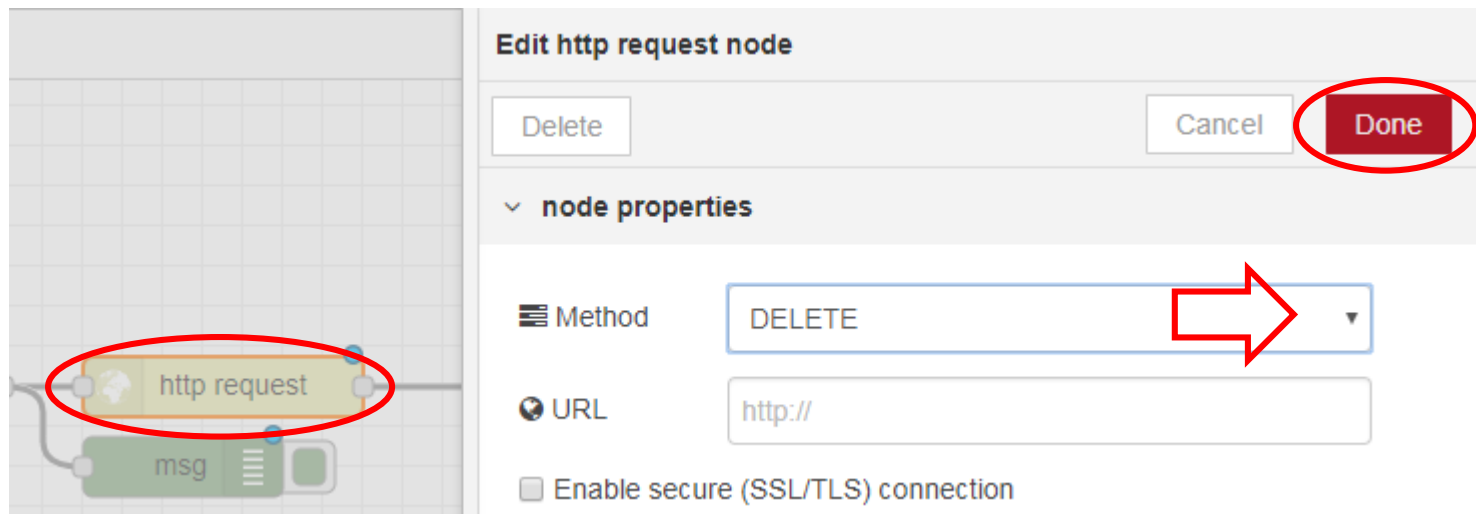
❖ NodeReD

❖ NodeRED

- Choose "http://localhost:8282" from the xN_CSE list
- Choose "MY_SENSOR_2" from the NM list
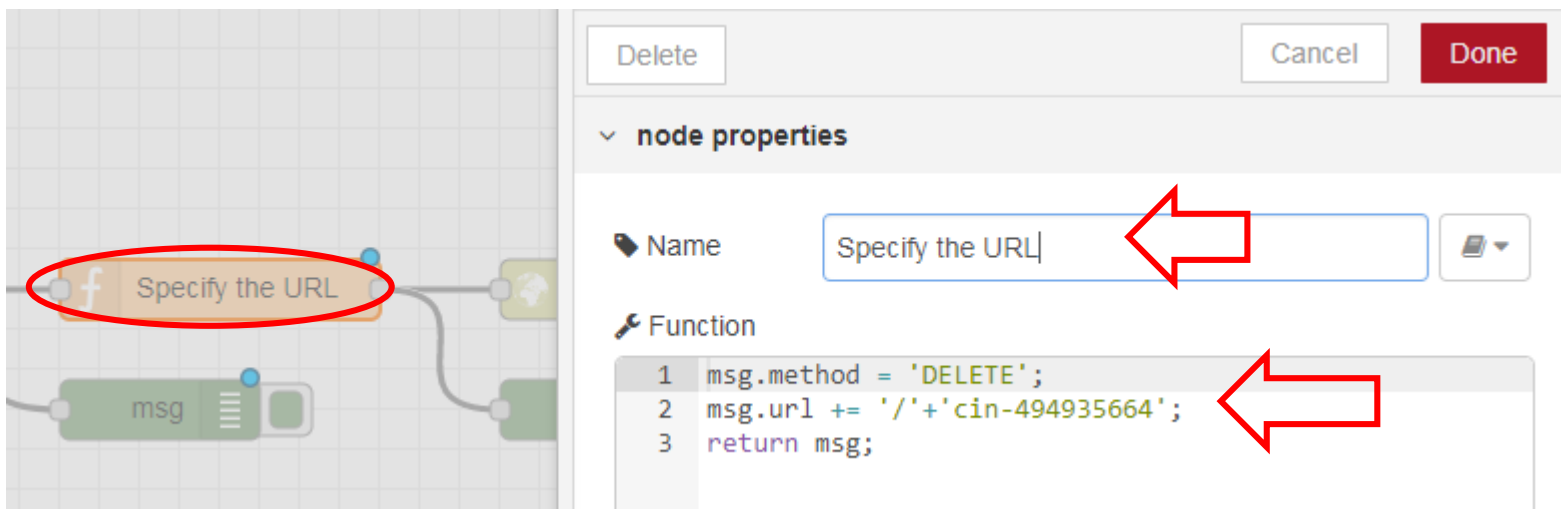
❖ NodeRED

- Choose "DELETE" from the Method list.

練習五：Delete

❖ NodeRED

- Add the details about "Specify the URL"


```
CODE
msg.method = 'DELETE';
msg.url += '/'+'資源的URN(去掉/mn-cse)';
return msg;
```

練習五：Delete

❖ **NodeRED**

- **Add the details about "Specify the URL"**
  - 由原先的URL中判斷：
    - 如果為*/in-cse則接上"in-name"
    - 如果為*/mn-cse則接上"mn-name"

CODE
```
msg.method = 'DELETE';
var mn_name = msg.url.match(/\/([^/-]*)-
[^/]*\/?$/);
if(mn_name!==null && mn_name.length>=2) {
    msg.url += '/' + mn_name[1] + '-
name/' + msg.headers['X-M2M-NM'];
}
return msg;
```

❖ This LAB shows how to use NodeRED to create web applications for OM2M.

- The OM2M plugin for NodeRED has been integrated into the provided VM.