

# JavaScript 物聯網實作

Node-RED 與網路爬蟲

王昱景 Brian Wang

brian.wang.frontline@gmail.com

# 安裝 Node-RED

# 安裝 Node.js



HOME | ABOUT | DOWNLOADS | DOCS | GET INVOLVED | SECURITY | NEWS



Node.js® is a JavaScript runtime built on [Chrome's V8 JavaScript engine](#).

Download for macOS (x64)

穩定版本

12.16.3 LTS

Recommended For Most Users

14.2.0 Current

Latest Features

最新版本

[Other Downloads](#) | [Changelog](#) | [API Docs](#) [Other Downloads](#) | [Changelog](#) | [API Docs](#)

Or have a look at the [Long Term Support \(LTS\) schedule](#).

Sign up for [Node.js Everywhere](#), the official Node.js Monthly Newsletter.



Report Node.js issue | Report website issue | Get Help

© OpenJS Foundation. All Rights Reserved. Portions of this site originally © Joyent.

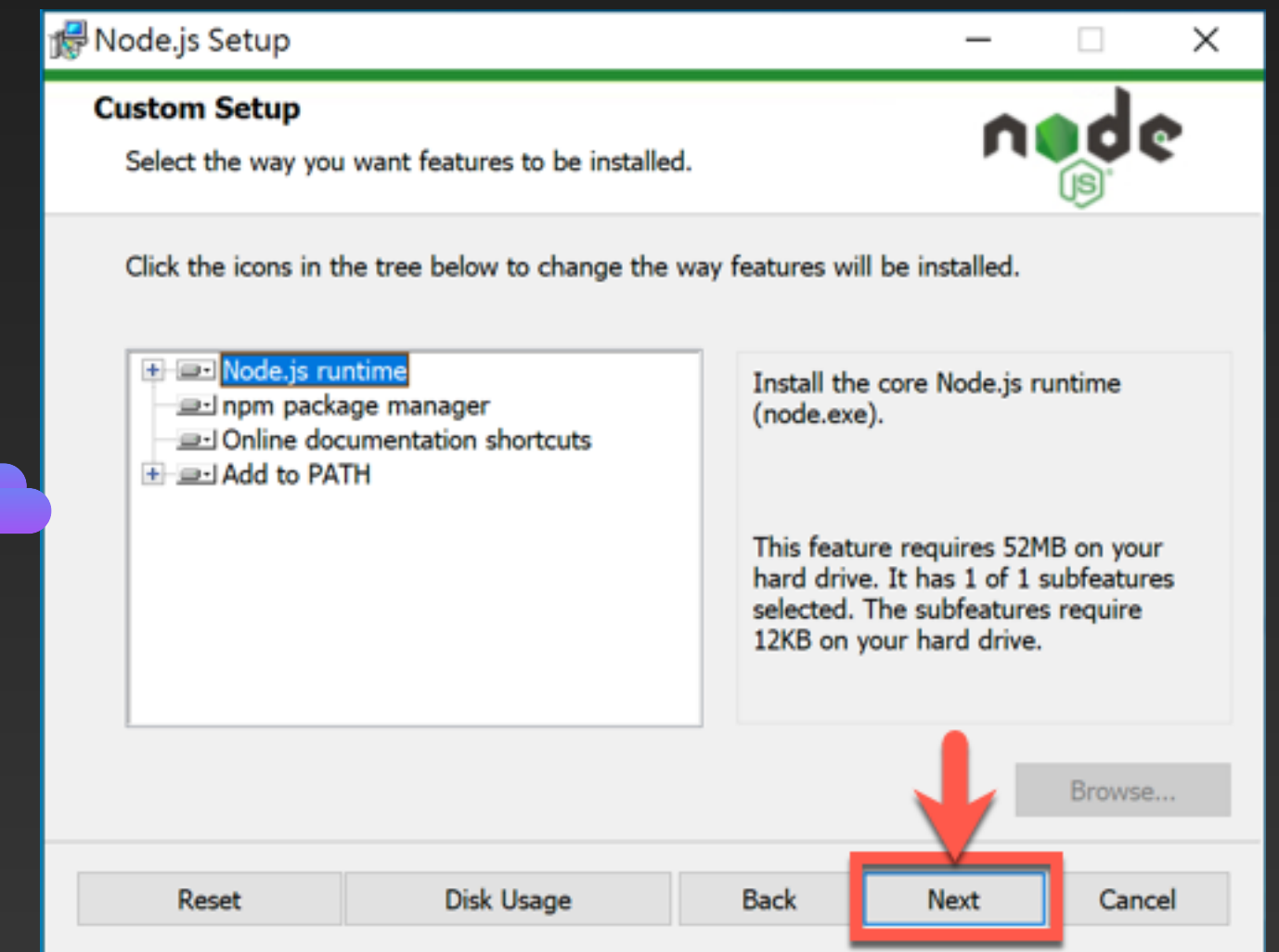
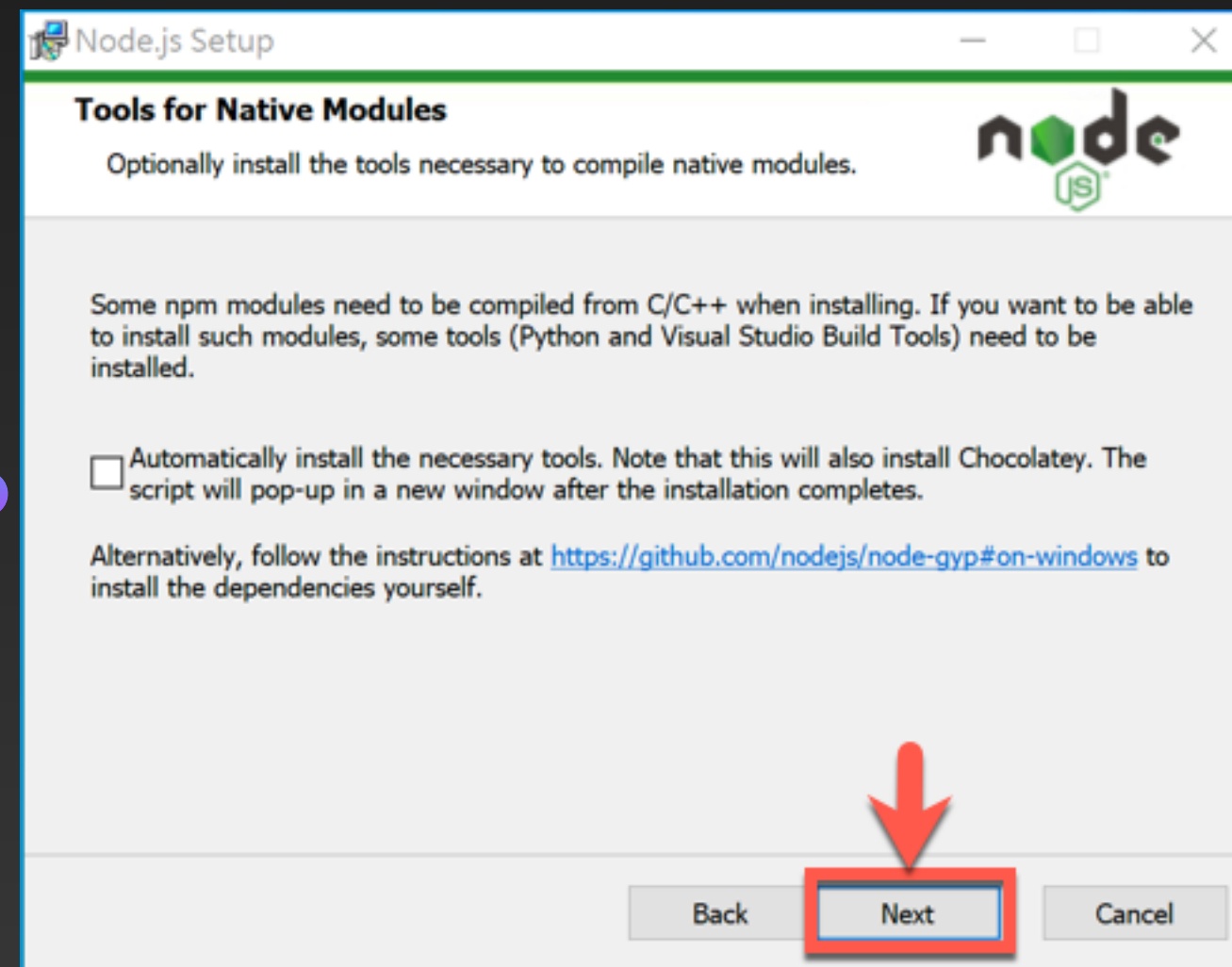
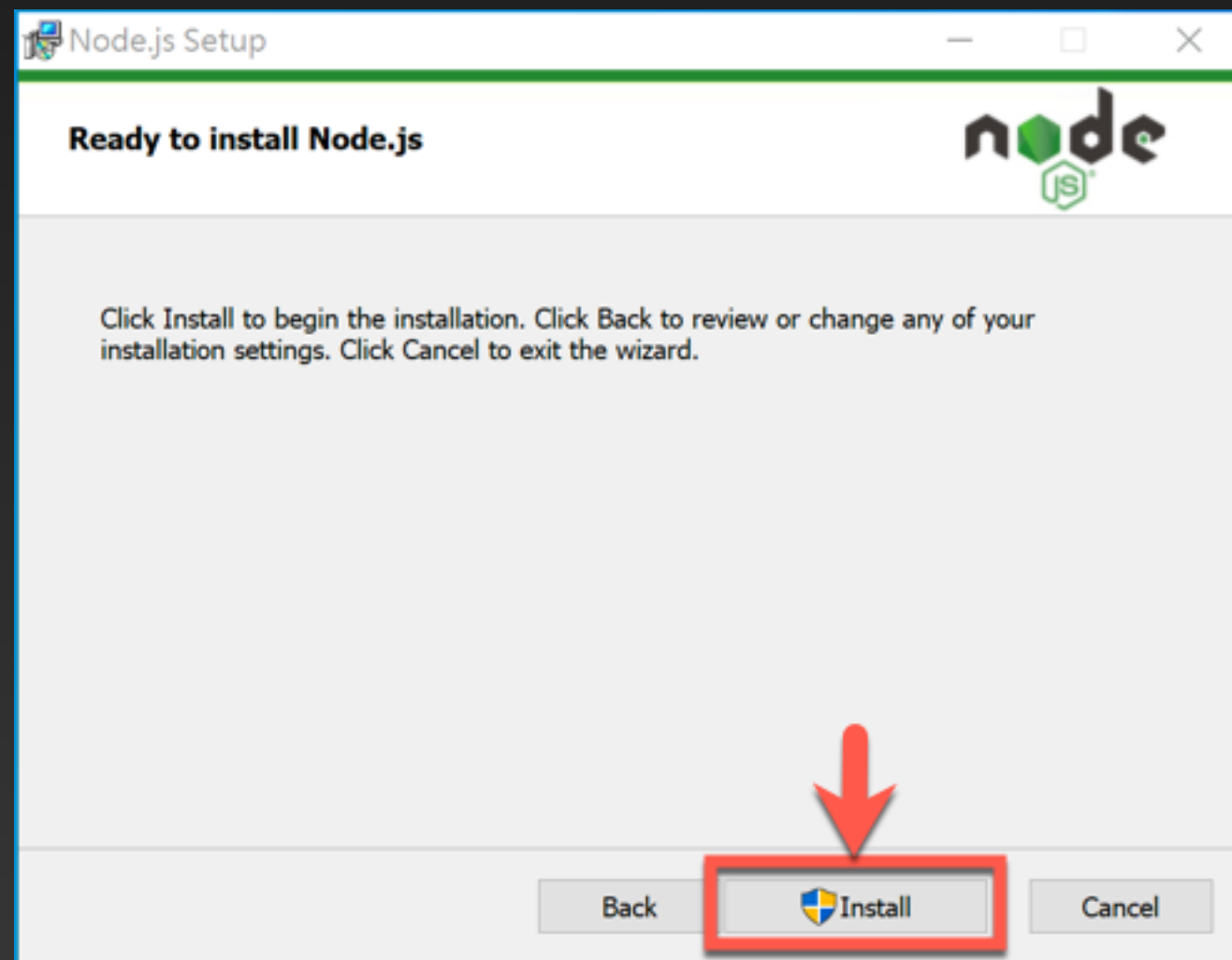
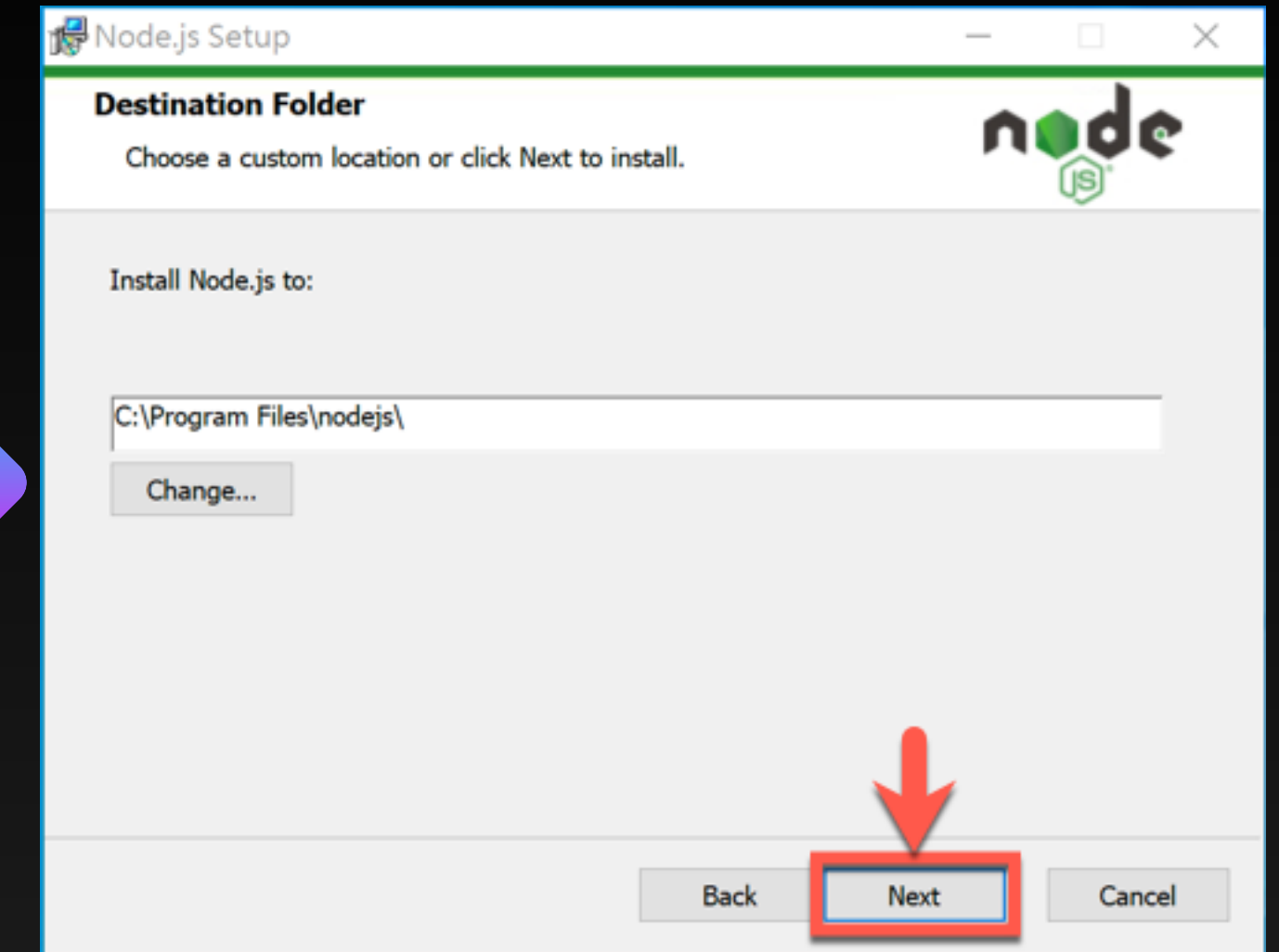
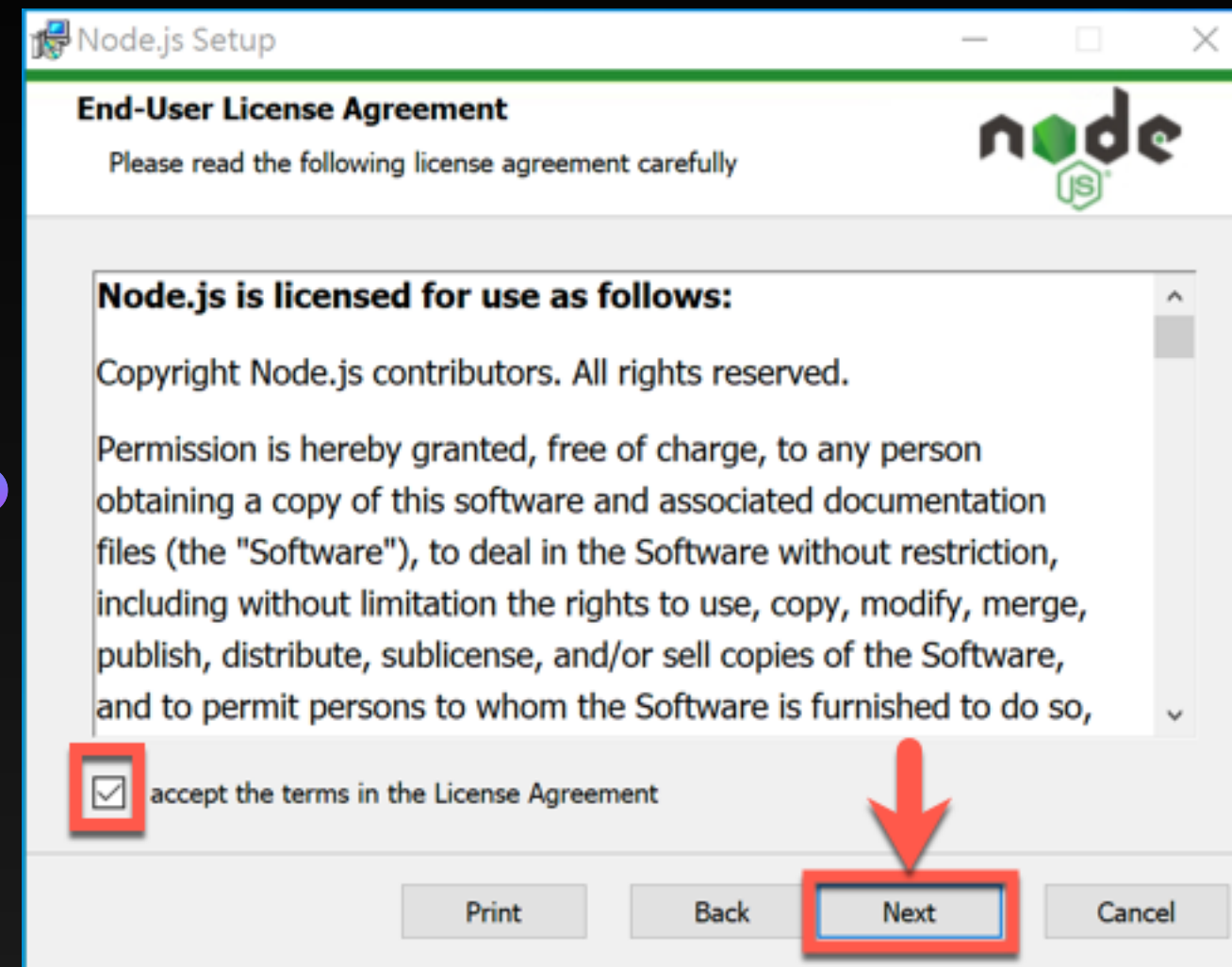
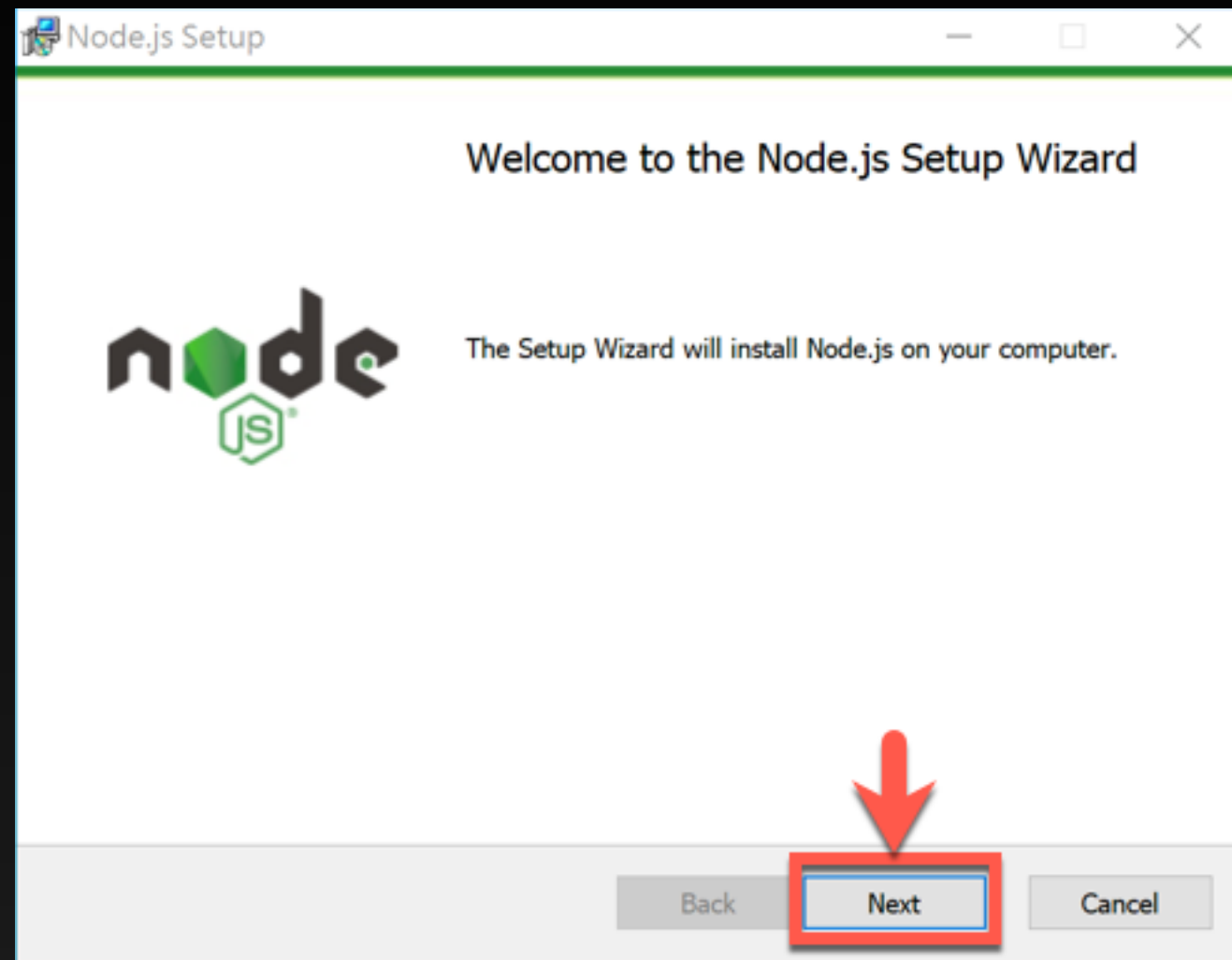
Node.js is a trademark of Joyent, Inc. and is used with its permission. Please review the Trademark Guidelines of the OpenJS Foundation.

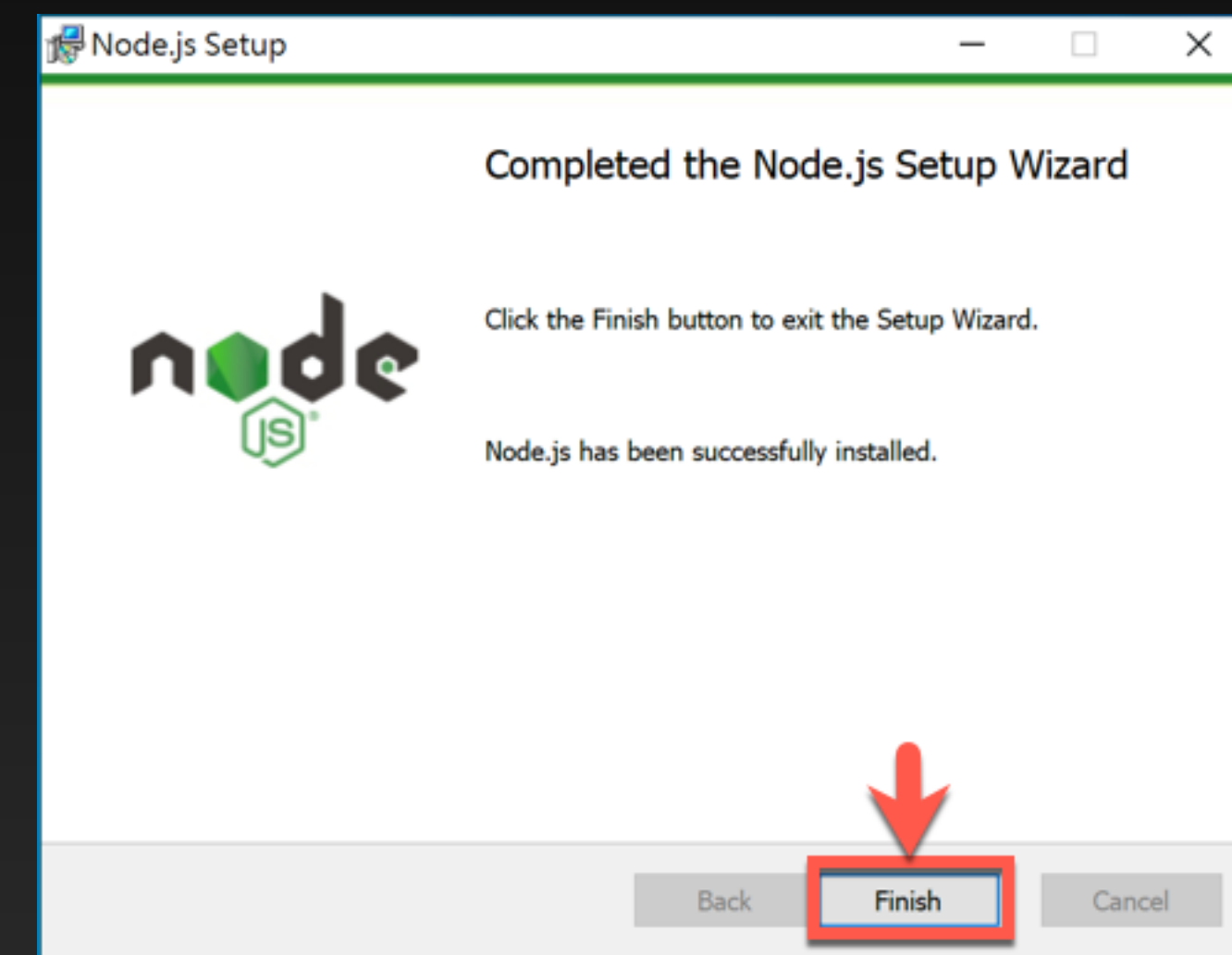
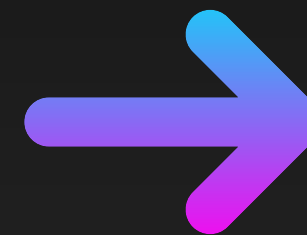
[Node.js Project Licensing Information](#).



Thank you drifkin for being a Node.js contributor 4 contributions

<https://nodejs.org/en/>





# 安裝 Node-RED

- 安裝 Node.js 完成後，打開終端機
- 輸入 `npm install -g node-red` 就可以完成 Node-RED 的安裝
- 如果是 Mac 前面要加上 `sudo`
- 安裝完成後，只要繼續輸入 `node-red`，就會啟動 Node-RED



Node.js command prompt

Your environment has been set up for using Node.js 12.17.0 (x64) and npm.

C:\Users\Brian Wang>npm install -g node-red

npm WARN deprecated request@2.88.0: request has been deprecated, see <https://github.com/request/request/issues/3142>

C:\Users\Brian Wang\AppData\Roaming\npm\node-red-pi -> C:\Users\Brian Wang\AppData\Roaming\npm\node\_modules\node-red\bin\node-red-pi

C:\Users\Brian Wang\AppData\Roaming\npm\node-red -> C:\Users\Brian Wang\AppData\Roaming\npm\node\_modules\node-red\red.js

> bcrypt@3.0.6 install C:\Users\Brian Wang\AppData\Roaming\npm\node\_modules\node-red\node\_modules\bcrypt

> node-pre-gyp install --fallback-to-build

node-pre-gyp WARN Using request for node-pre-gyp https download

[bcrypt] Success: "C:\Users\Brian Wang\AppData\Roaming\npm\node\_modules\node-red\node\_modules\bcrypt\lib\binding\bcrypt\_lib.node" is installed via remote

+ node-red@1.0.6

added 330 packages from 339 contributors in 46.36s

C:\Users\Brian Wang>

啟動 Node-RED



- 打開終端機輸入 `node-red` 就會啟動 Node-RED
- 打開瀏覽器，網址列輸入 `http://127.0.0.1:1880/`，就可以打開 Node-RED
- 左邊是一些功能流程的節點，每個節點都有各自的功能
- 中間的區域就是透過這些節點組成流程圖

```
node-red
C:\Users\Brian Wang>node-red
27 May 21:19:26 - [info]

Welcome to Node-RED
=====

27 May 21:19:26 - [info] Node-RED version: v1.0.6
27 May 21:19:26 - [info] Node.js version: v12.17.0
27 May 21:19:26 - [info] Windows_NT 10.0.18362 x64 LE
27 May 21:19:27 - [info] Loading palette nodes
27 May 21:19:28 - [info] Settings file : C:\Users\Brian Wang\.node-red\settings.js
27 May 21:19:28 - [info] Context store : 'default' [module=memory]
27 May 21:19:28 - [info] User directory : C:\Users\Brian Wang\.node-red
27 May 21:19:28 - [warn] Projects disabled : editorTheme.projects.enabled=false
27 May 21:19:28 - [info] Flows file : C:\Users\Brian Wang\.node-red\flows_DESKTOP-9TIQAEg.json
27 May 21:19:28 - [info] Creating new flow file
27 May 21:19:28 - [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.
-----

27 May 21:19:28 - [info] Server now running at http://127.0.0.1:1880/
27 May 21:19:28 - [info] Starting flows
27 May 21:19:28 - [info] Started flows
```

Node-RED

127.0.0.1:1880/#flow/be2808c7.681ff8

Node-RED

部署

過濾節點

流程1

共通

inject

debug

complete

catch

status

link in

link out

comment

功能

網絡

序列

解析

存儲

信息

信息

流程	"be2808c7.681ff8"
名稱	流程1
狀態	啟用

描述

按住↑的同時點擊節點會選中所有被連接的節點。

# Node-RED 爬蟲 (日幣匯率)

<https://hackmd.io/@ycwang812/B1H2HUB9L>



# 解析匯率網頁

台灣銀行的牌告匯率 (<http://rate.bot.com.tw/xrt?Lang=zh-TW>)

 臺灣銀行 BANK OF TAIWAN						
2017/01/09 本行營業時間牌告匯率						
<p>請注意：</p> <ol style="list-style-type: none"><li>1. 本表資料僅供參考，不代表實際交易匯率。</li><li>2. 「網路銀行」及「Easy購線上申購現鈔或旅支」之實際交易匯率，以交易時顯示之匯率為準。</li><li>3. 臨櫃實際交易匯率以交易時本行匯率為準。</li><li>4. 本網頁牌告匯率資訊為靜態顯示，顯示之牌告匯率資訊不會隨後續異動而自動更新資訊，欲得知本行最新牌告匯率資訊請按「取得最新報價」鈕。</li></ol>						
<div>取得最新報價</div> <div>線上申購外幣現鈔或旅支</div>						
牌價最新掛牌時間：2017/01/09 16:00						
幣別	現金匯率		即期匯率		遠期匯率	歷史匯率
	本行買入	本行賣出	本行買入	本行賣出		
 美金 (USD)	31.75	32.292	32.05	32.15	查詢	查詢
 港幣 (HKD)	3.989	4.184	4.109	4.169	查詢	查詢
 英鎊 (GBP)	38.05	39.98	38.92	39.34	查詢	查詢
 澳幣 (AUD)	23.17	23.83	23.36	23.59	查詢	查詢
 加拿大幣 (CAD)	23.83	24.57	24.1	24.32	查詢	查詢
 新加坡幣 (SGD)	21.76	22.54	22.18	22.36	查詢	查詢
 瑞士法郎 (CHF)	30.84	31.9	31.37	31.66	查詢	查詢
 日圓 (JPY)	0.2651	0.2761	0.2715	0.2755	查詢	查詢
 南非幣 (ZAR)	-	-	2.29	2.37	查詢	查詢



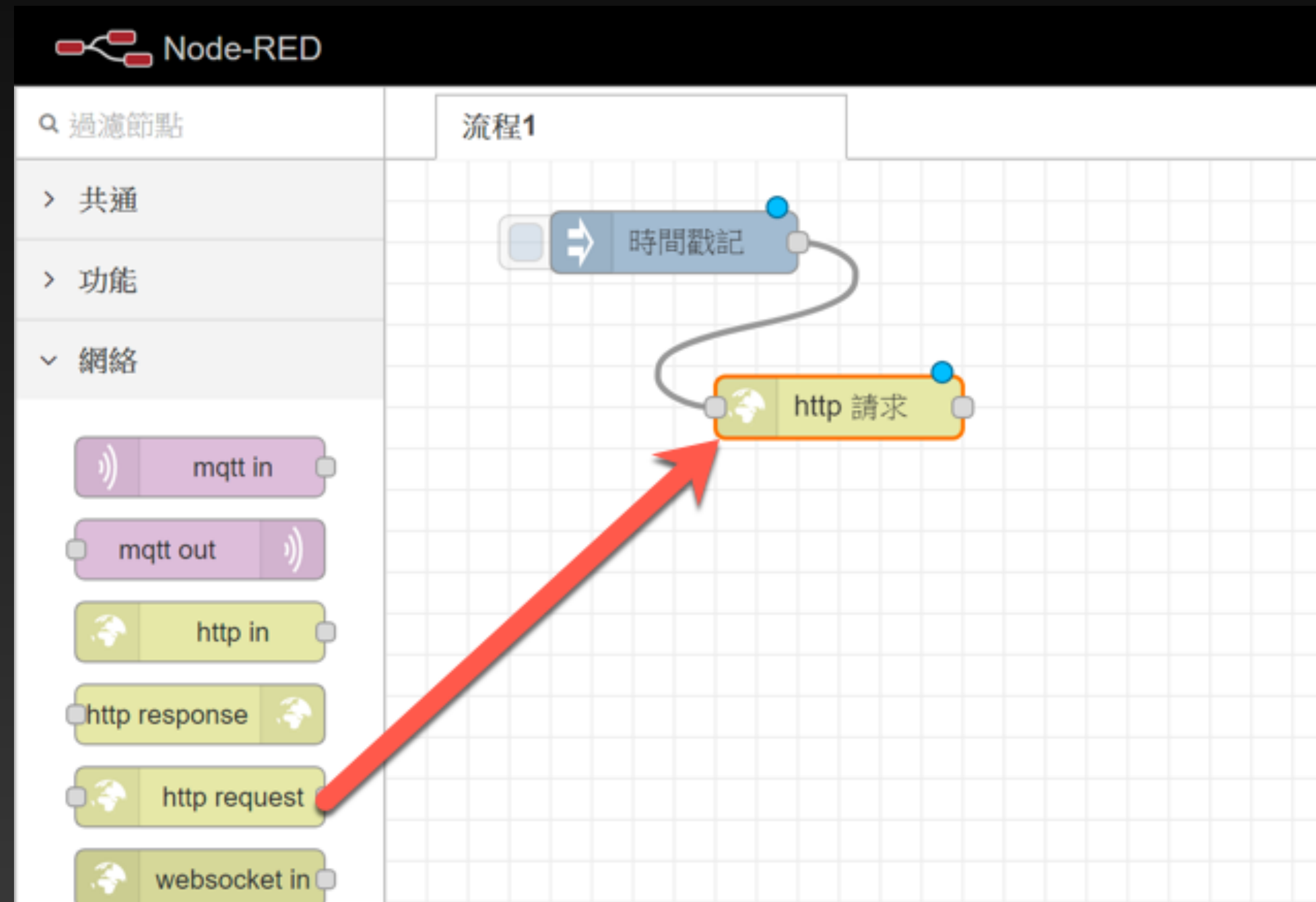




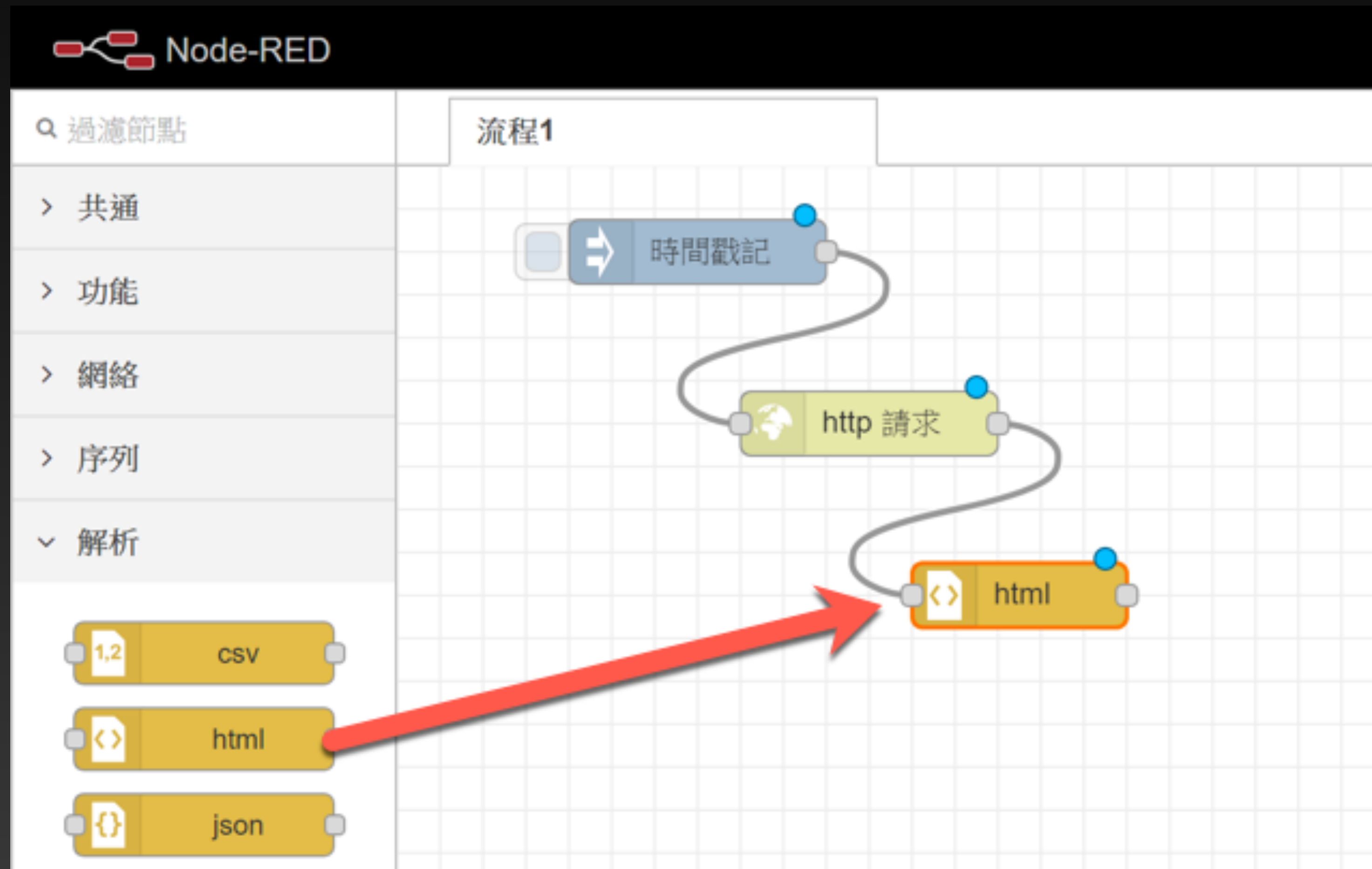
- 把這段 class 複製下來，解析網頁時需要抓取的名稱  
rate-content-cash text-right print\_hide

```
<td data-table="本行現金買入" class="rate-content-cash text-right print_hide">0.2679</td>  
<td data-table="本行現金賣出" class="rate-content-cash text-right print_hide">0.2789</td>  
... "十二個月定期存款" ...
```

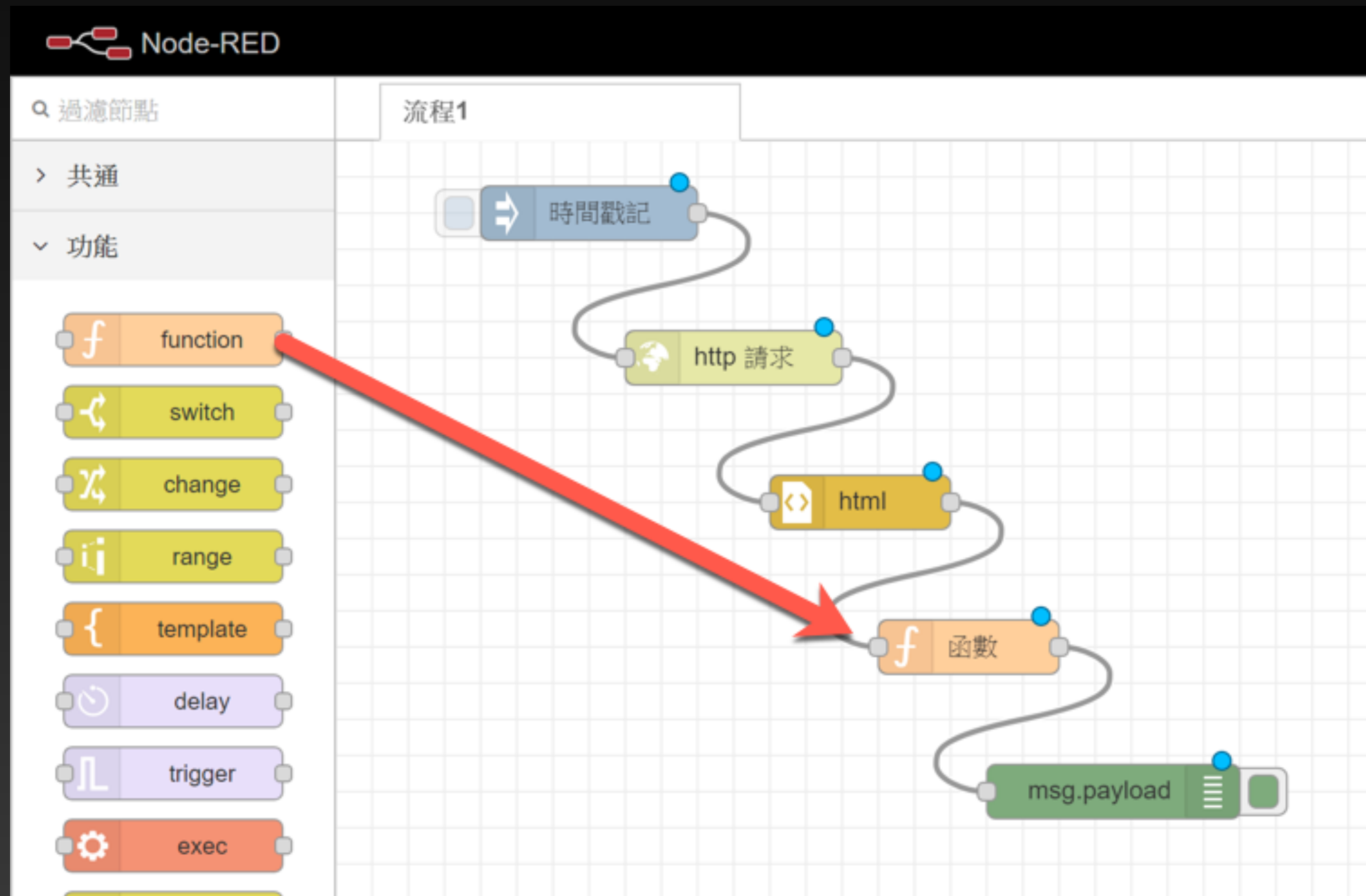
把 inject 節點放進去，然後把 http request 放在之後



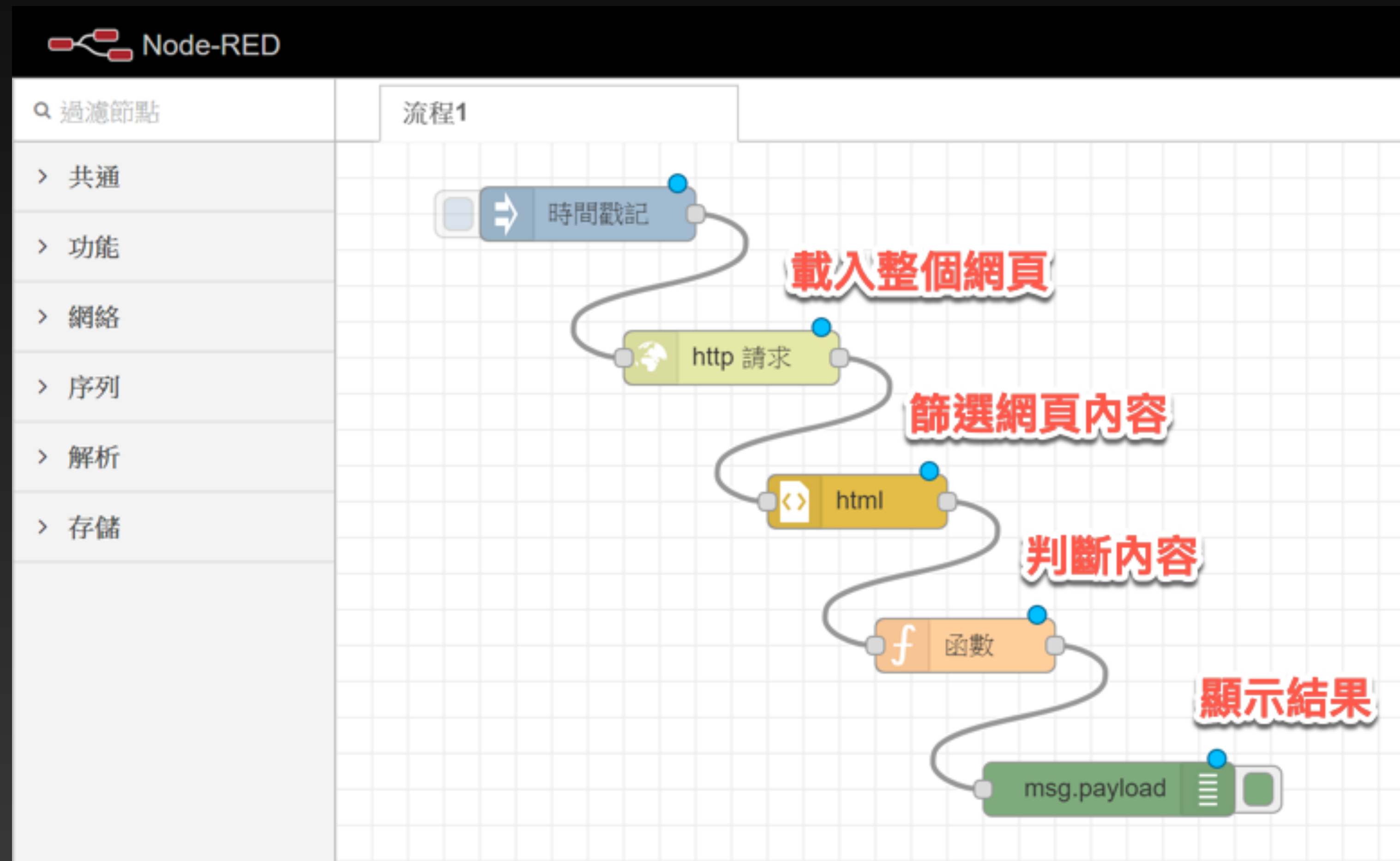
## 在 http 請求結點後加上 html 節點



## 最後加上 function 及 debug 節點



透過 http request 抓取整個牌告匯率網頁，然後透過 html 節點篩選出特定內容，由 function 節點做分析，最後將結果由 debug 顯示出來



點選 http 請求打開設定，於 URL 填入牌告匯率網址 <http://rate.bot.com.tw/xrt?Lang=zh-TW>

編輯http request節點

刪除 取消 完成

⚙ 屬性

☰ 請求方式 GET

🌐 URL <http://rate.bot.com.tw/xrt?Lang=zh-TW>

☐ 將msg.payload附加為查詢字符串參數

☐ 使用安全連接 (SSL/TLS)

☐ 基本認證

☐ 對連接啟用keep-alive

☐ 使用代理服務器

← 返回 UTF-8 字串

🏷 名稱 名稱



點選 html 打開設定，選取項填入 .rate-content-cash.text-right.print\_hide，注意這邊的選取項是 CSS 選擇器，輸出選擇選定元素的 html 內容，名稱為 filter

編輯html節點

刪除 取消 完成

⚙ 屬性

⋮ 屬性 msg. payload

🔍 選取項 **.rate-content-cash.text-right.print\_hide**

🔗 輸出 選定元素的html內容

一條資訊 [陣列]

in msg. payload

🏷 名稱 filter

先部署點選 inject，結果呈現一組陣列

這組陣列表示整份網頁裡面 .rate-content-cash.text-right.print\_hide 的值依序排列，而要的日幣匯率在第 15 位

The screenshot displays a workflow editor on the left and a console window on the right. The workflow, titled '流程1', consists of five steps: '時間戳記' (Timestamp), 'http 請求' (HTTP Request), 'filter', 'f 函數' (Function), and 'msg.payload' (Message Payload). The console window, titled '除錯窗口' (Debug Console), shows the output of the workflow at 2020/6/1 下午9:36:32. The output is an array of 38 elements, with the 14th and 15th elements highlighted in a red box. The 14th element is '0.2691' and the 15th element is '0.2819'.

```
2020/6/1 下午9:36:32 node: 6aeee776.b3ffb8
msg.payload : array[38]
  ▼ array[38]
    ▶ [0 ... 9]
    ▼ [10 ... 19]
      10: "20.71"
      11: "21.62"
      12: "30.37"
      13: "31.57"
      14: "0.2691"
      15: "0.2819"
      16: "-"
      17: "-"
      18: "2.81"
      19: "3.33"
    ▶ [20 ... 29]
    ▶ [30 ... 37]
```

點選函數打開設定，輸入下面程式碼，顯示的結果會是「( 時:分:秒 ) 日幣匯率：....」

```
var jp = msg.payload[15];  
var date = new Date();  
var h = date.getHours();  
var m = date.getMinutes();  
var s = date.getSeconds();
```

```
if (h < 10) {  
    h = '0' + h;  
}  
if (m < 10) {  
    m = '0' + m;  
}  
if (s < 10) {  
    s = '0' + s;  
}
```

```
msg.payload = '(' + h + ':' + m + ':' + s + ')\n' + '日幣匯率：' + jp ;
```

```
return msg;
```

## 編輯function節點

刪除

取消

完成

⚙ 屬性



🏷 名稱

名稱



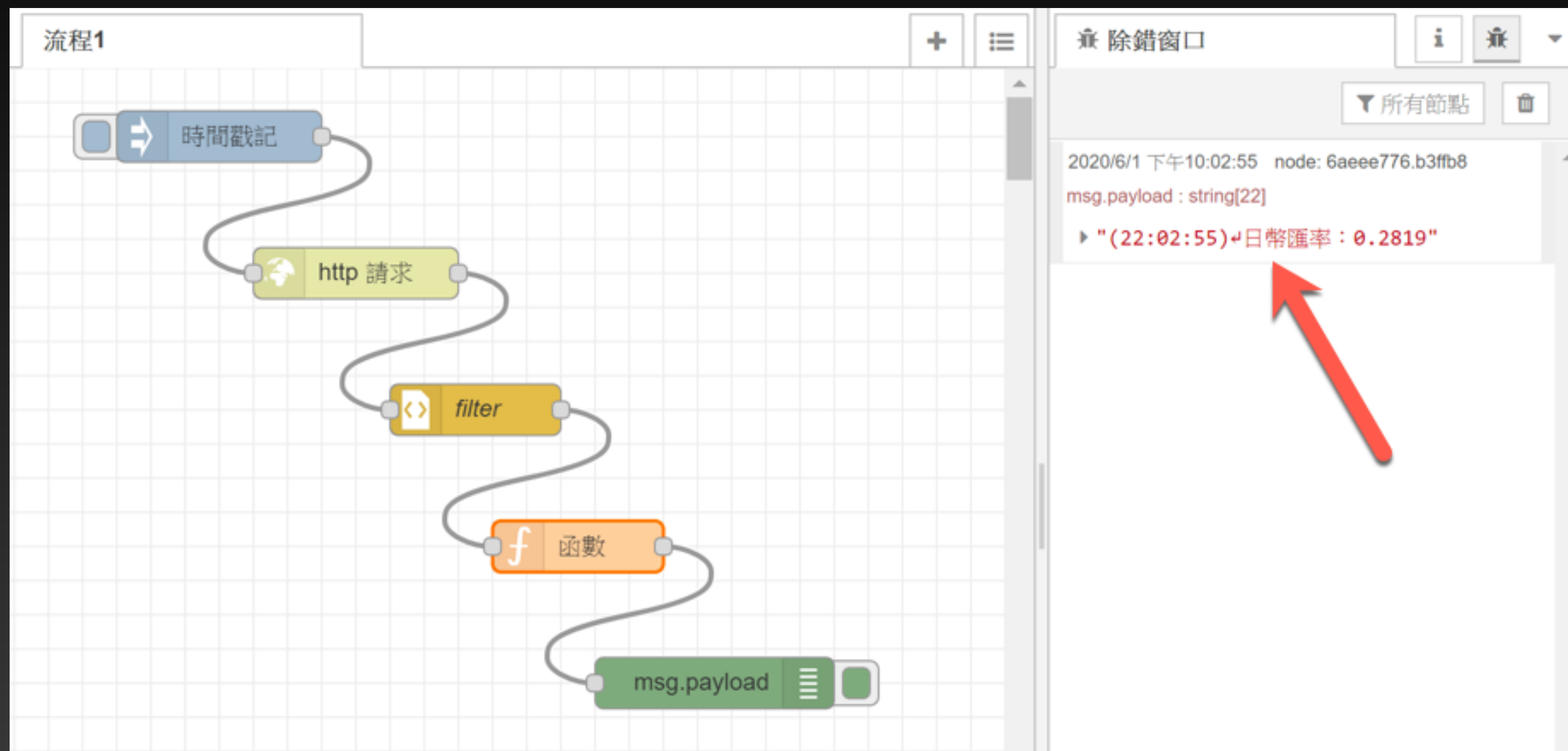
🔧 函數



第15位

```
1 var jp = msg.payload[15];
2 var date = new Date();
3 var h = date.getHours();
4 var m = date.getMinutes();
5 var s = date.getSeconds();
6
7 if (h < 10) {
8     h = '0' + h;
9 }
10
11 if (m < 10) {
12     m = '0' + m;
13 }
14
15 if (s < 10) {
16     s = '0' + s;
17 }
18
19 msg.payload = '(' + h + ':' + m + ':' + s + ')\n'+
20 '日幣匯率:' + jp ;
21
22 return msg;
```

完成後將流程部署，點選時間戳記，可以看到即時的匯率出現



講義、範例程式下載：

- <http://ycwang812.at.tw>

Repositories > UCH

- <https://github.com/ycwang812/UCH>

