

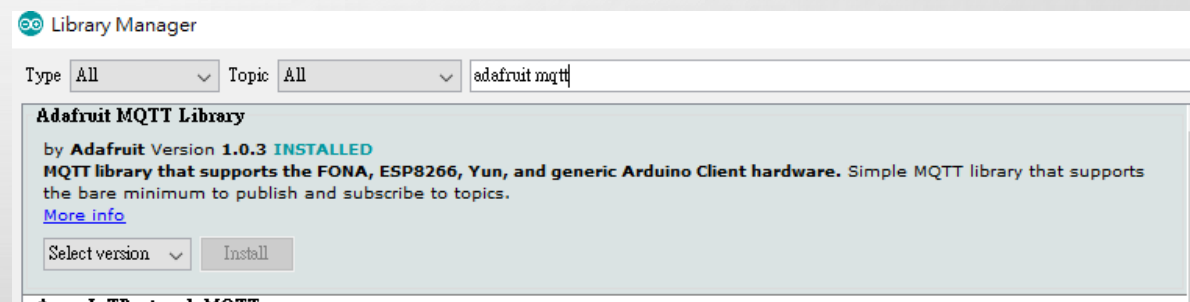
The background is a light gray gradient. It is decorated with numerous realistic water droplets of various sizes, some with highlights and shadows, scattered across the surface. In the upper center, there is a faint, circular, embossed-style logo that appears to be a stylized 'A' or a similar geometric shape.

ESP8266 + ADAFRUIT.IO筆記

楊鴻翊

需要LIBRARY

- esp8266 (在Boards Manager裡面安裝)
- adafruit MQTT Library (在Library Manager裡面安裝)



連結方式

- 1.透過MQTT (一種用在硬體性能受限的消息傳遞協議)
- 2.透過IFTTT(一個網路平台(全名"**IF This Then That**"), 可以用來當訊息的中繼站)

透過MQTT

- 使用adafruit MQTT Library
- 使用<https://github.com/shooter2062424/ArduinoFunction/tree/master/MQTT> 的包裝
 - 將Adafruit MQTT包裝在Connector class中
 - 以下將透過範例code來解說

透過MQTT

- 在ADAFRUIT.IO上建立帳號
- 設定範例程式的資料
- 建立儀錶板
- 建立變數
- 在儀錶板上顯示變數

透過MQTT – 範例程式需要額外填寫的必要資訊(1/2)

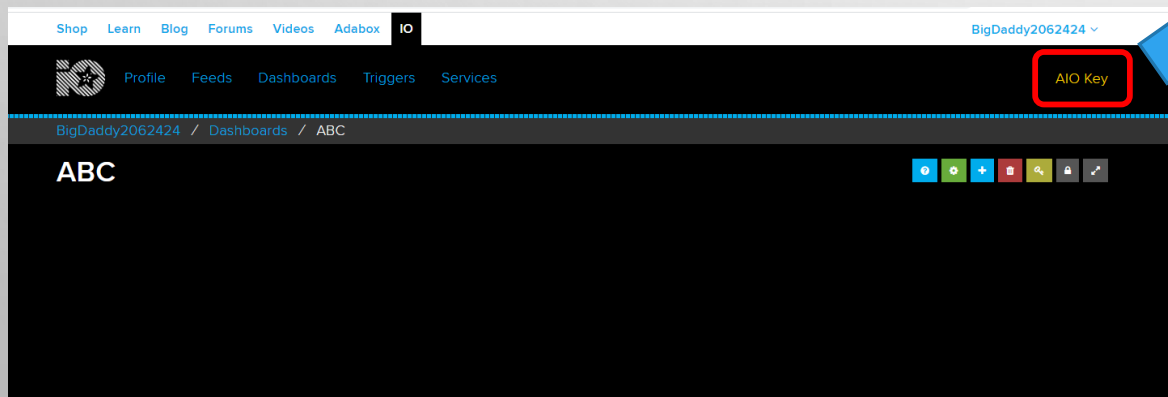
```
18 #define WLAN_SSID
19 #define WLAN_PASS
20
21 /*****
22
23 #define AIO_SERVER
24 #define AIO_SERVERPORT
25 #define AIO_USERNAME
26 #define AIO_KEY
```

WLAN_SSID : 輸入欲連接wifi的名稱

WLAN_PASS : 輸入欲連接wifi的密碼

AIO_USERNAME : adafruit.io的帳號名稱

AIO_KEY : adafruit.io的秘密金鑰



YOUR AIO KEY

Your Adafruit IO key should be kept in a safe place and treated with the same care as your Adafruit username and password. People who have access to your AIO key can view all of your data, create new feeds for your account, and manipulate your active feeds.



If you need to regenerate a new AIO key, all of your existing programs and scripts will need to be manually changed to the new key.

AIO_USERNAME

Username

BigDaddy2062424

AIO_KEY

Active Key

aio_

REGENERATE AIO KEY

透過MQTT – 範例程式需要額外填寫的必要資訊(2/2)

```
47 mqtt->Subscribe("BigDaddy2062424/feeds/Toggle", rcvValue);
```

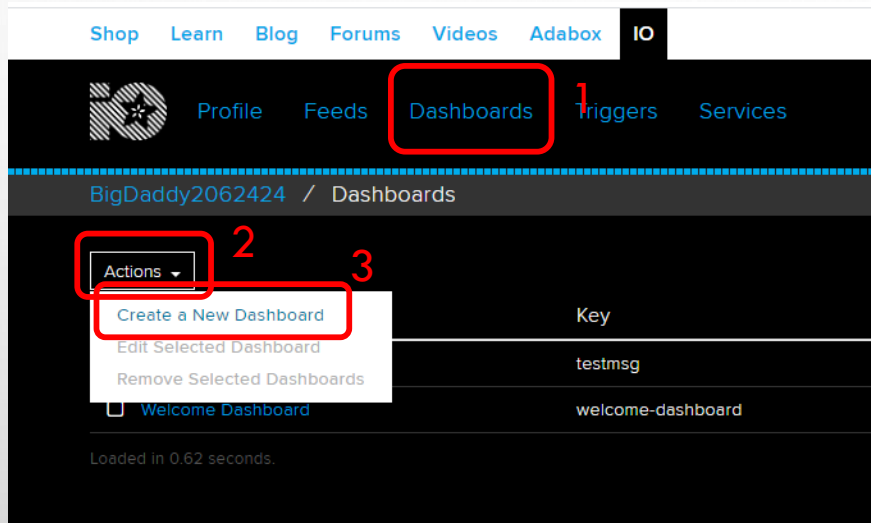
範例程式第47行藍色字串：訂閱的訊息

```
61 if (!mqtt->Publish("BigDaddy2062424/feeds/Value", 101)) {
```

範例程式第61行藍色字串：傳送的訊息

稍後設定完Adafruit.io的介面後就會有自己的訊息!

透過MQTT – 建立儀錶板

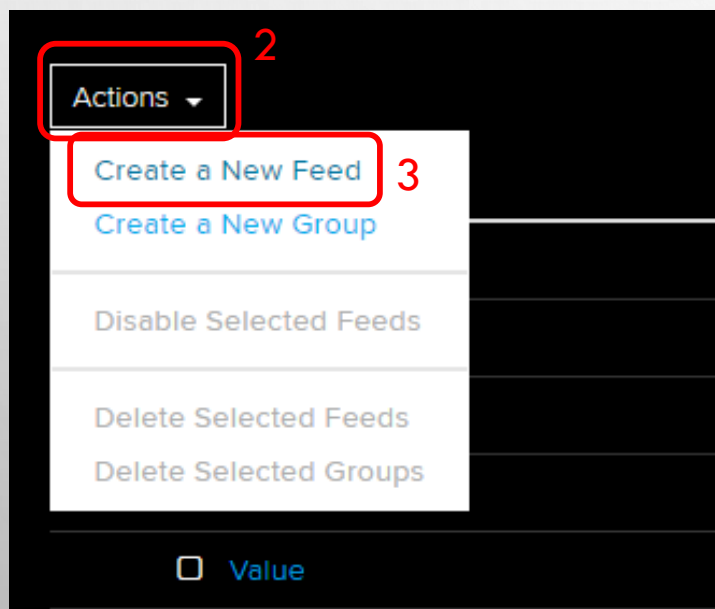
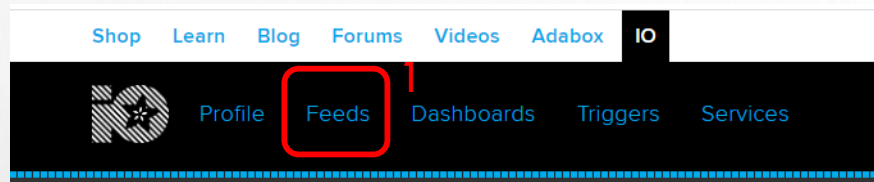


- 1.點選上方Dashboards標籤
- 2.點選Actions
- 3.點選Create a New Dashboard
- 4.輸入儀錶板名稱
- 5.點選Create建立
- 6.檢查是否建立成功

This screenshot shows the 'Create a new Dashboard' modal form. It has a title bar with a close button (X). The form contains two input fields: 'Name' and 'Description'. The 'Name' field is highlighted with a red box and labeled '4', and it contains the text 'ABC'. The 'Description' field is a larger text area below it. At the bottom right, there are two buttons: 'Cancel' and 'Create'. The 'Create' button is highlighted with a red box and labeled '5'.



透過MQTT – 建立變數



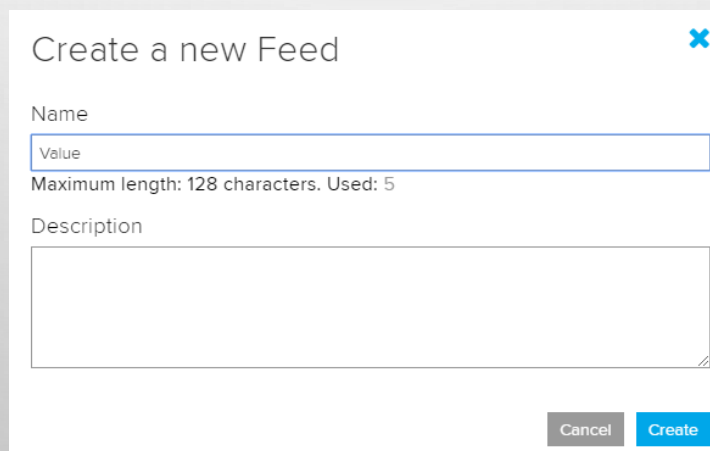
1.點選Feeds

2.點選Actions

3.點選Create a New Feed

4.接著建立Value變數

5.重複3、4建立Toggle變數



Create a new Feed

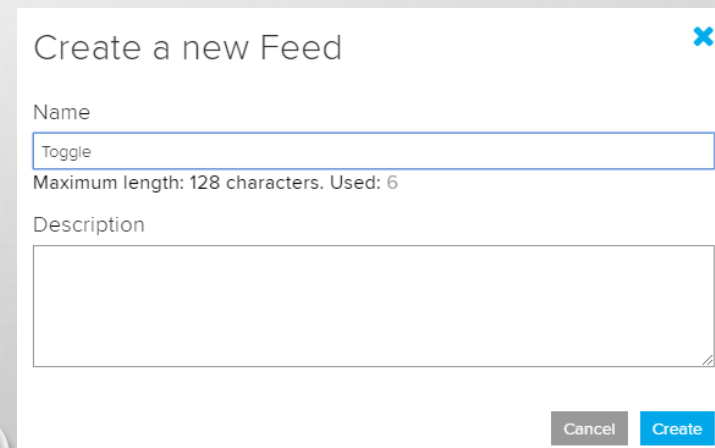
Name

Value

Maximum length: 128 characters. Used: 5

Description

Cancel Create



Create a new Feed

Name

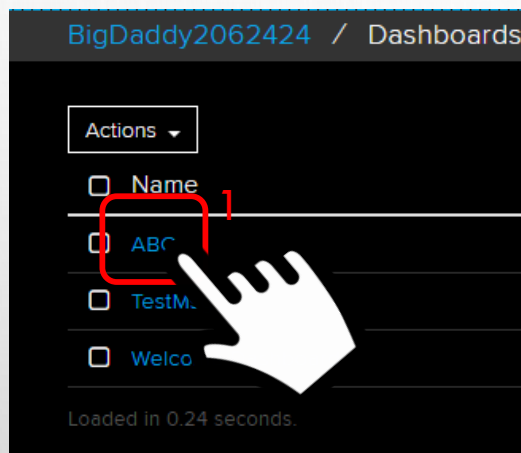
Toggle

Maximum length: 128 characters. Used: 6


Description

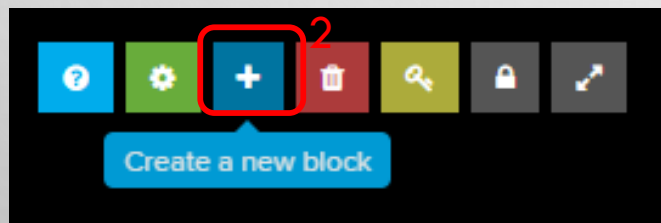
Cancel Create

透過MQTT –在儀錶板上面顯示變數(1 / 3)

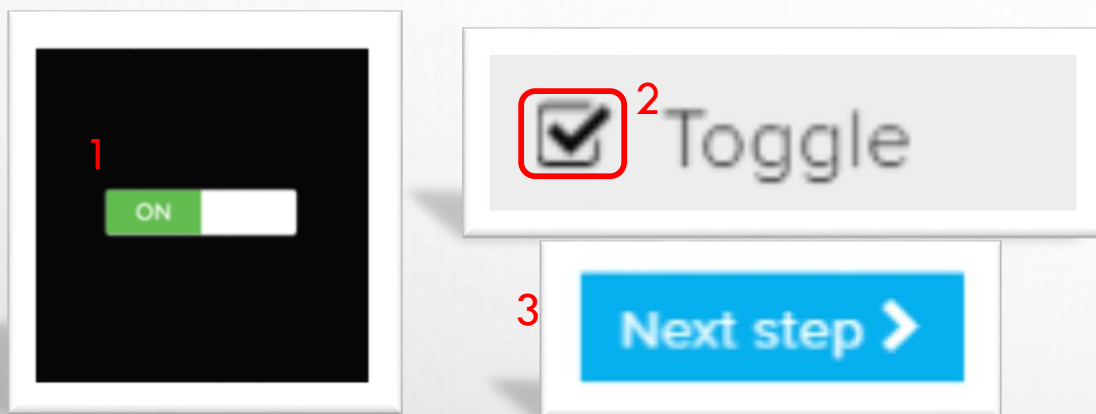


1.回到Dashboards並且點選剛剛建立的儀表板

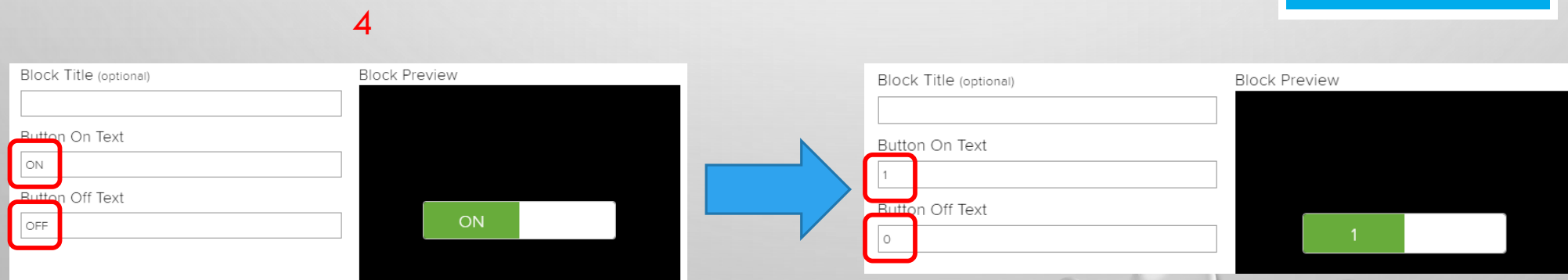
2.點選  來建立新的UI



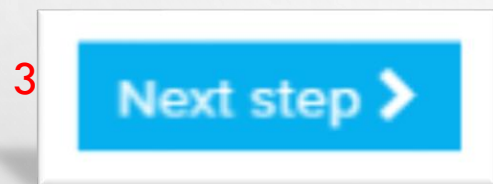
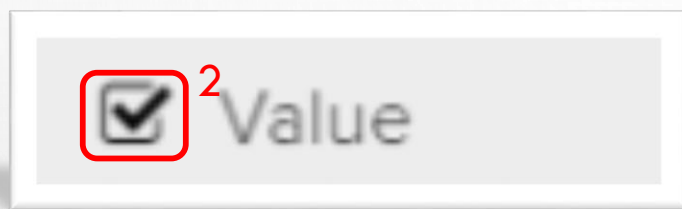
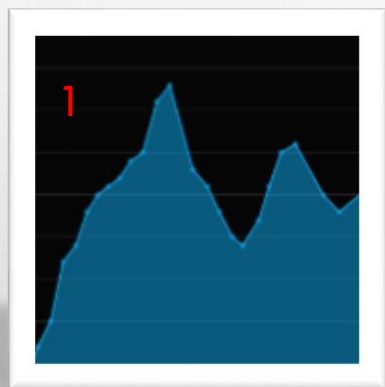
透過MQTT –在儀錶板上顯示變數(2/3)



- 1.點選左圖類似開關的按鍵
- 2.將Toggle變數左邊的框框點選勾起來
- 3.點選Next step進入下一步
- 4.將開關的資訊更改為1(On)與0(Off)
- 5.點選Create block完成建立

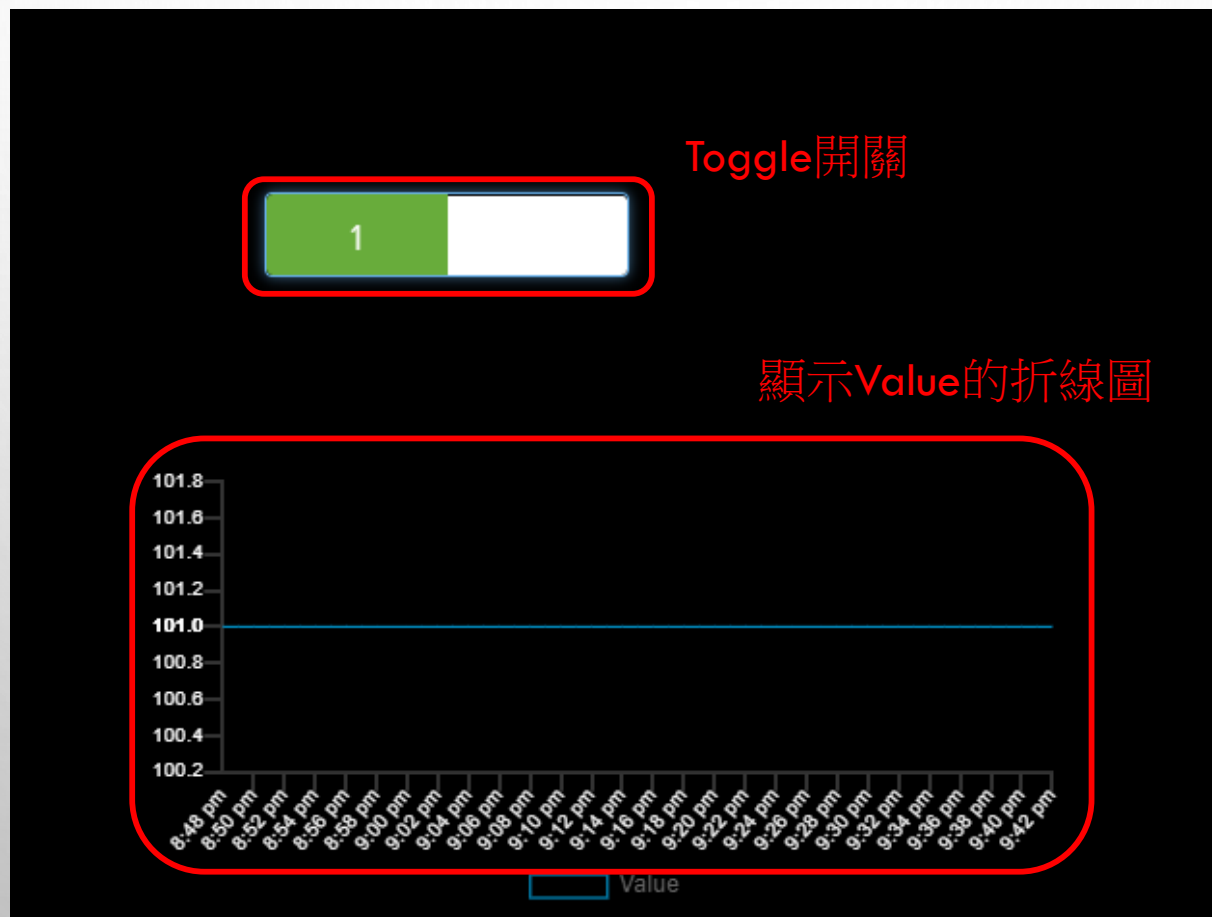


透過MQTT –在儀錶板上面顯示變數(3/3)



- 1.點選左圖類似折線圖的按鍵
- 2.將Value變數左邊的框框點選勾起來
- 3.點選Next step進入下一步
- 4.點選Create block完成建立

透過MQTT – 完成後的DASHBOARD介面



透過MQTT – 訂閱、發布訊息的ID

Adafruit的訊息ID格式為

<使用者帳戶>/feeds/<變數名稱>

因此本範例第47行與第61行該填入的為

<你的Adafruit使用者帳戶名稱>/feeds/Toggle

<你的Adafruit使用者帳戶名稱>/feeds/Value

此外Adafruit官方有提供的時間訊息為

1. time/seconds : 以秒當作單位
2. time/millis : 以毫秒當作單位
3. time/hours : 以小時當作單位

透過MQTT – 執行程式

- 跑起來!
- 看程式的註解來觀察怎麼使用~

透過IFTTT

- 使用IFTTT網站設定訊息的來源與該送去哪
- 好麻煩Q 晚點再看看怎麼搞