

手把手帶你用Line Notify

NiJia

林家燈 / Nijia

凱納股份有限公司 - Backend Developer

朝陽科技大學 - 碩士

COSCUP 2019 chatbot - 負責人

Chatbot TW meetup - 講者

圖奇獸 chatbot - 作者

首先先來個 ps 指令

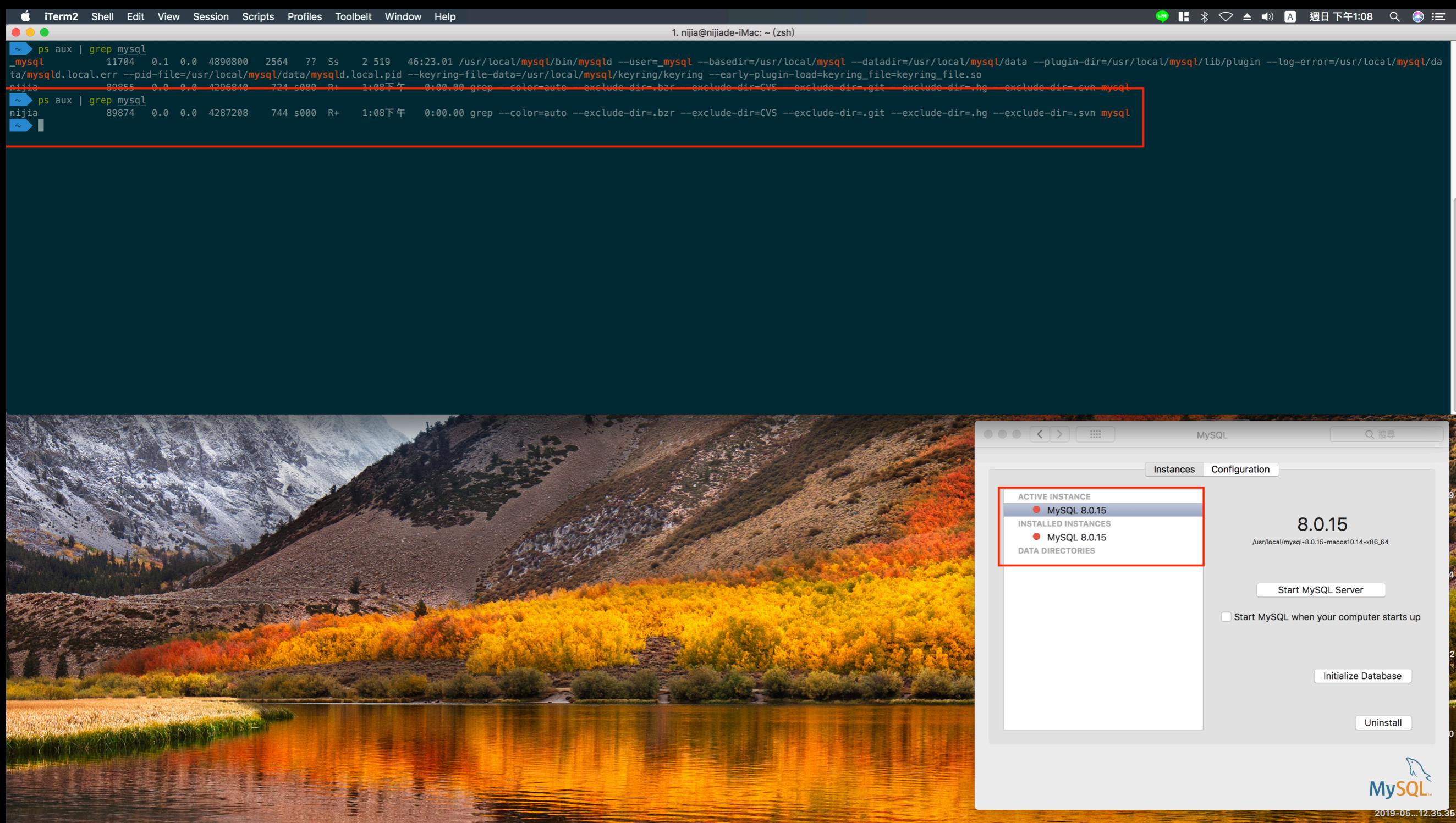
```
ps aux | grep mysql
```

系統偏好設定 編輯 顯示方式 視窗 輔助說明

1. niiia@nijiade-iMac: ~ (zsh)

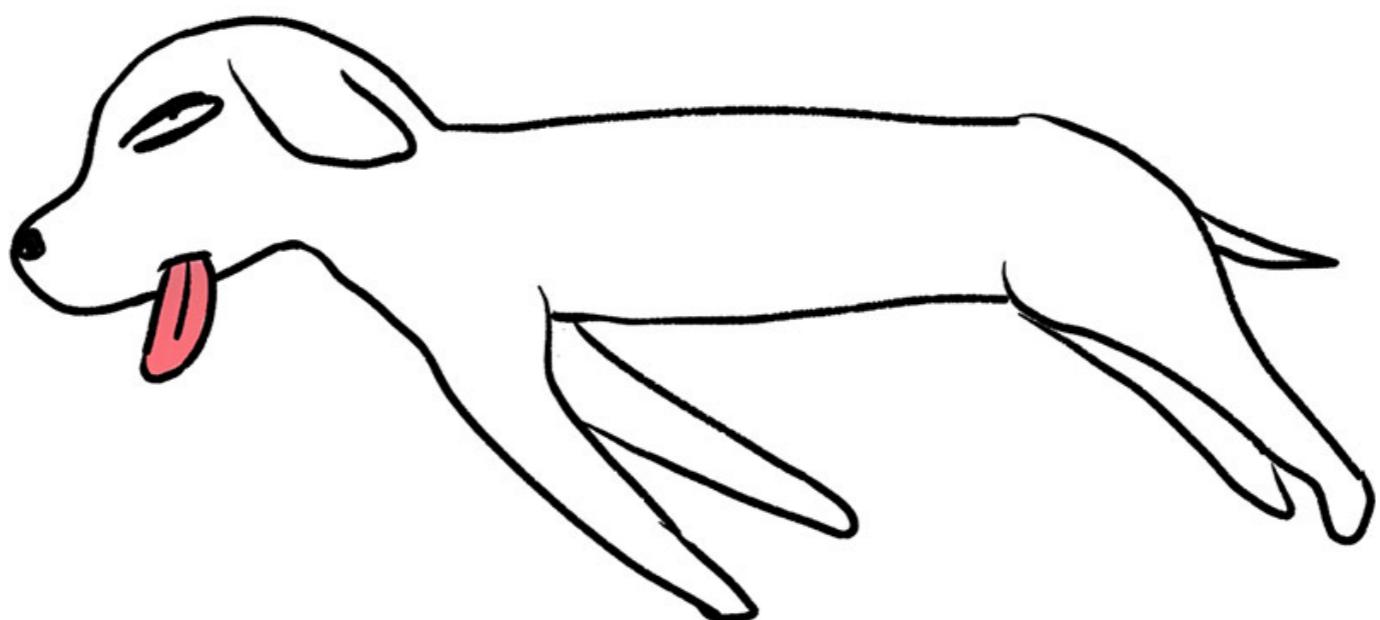
```
~▶ ps aux | grep mysql
mysql      11704  0.1  0.0  4890800   2564 ?? Ss    2 519  46:23.01 /usr/local/mysql/bin/mysqld --user=_mysql --basedir=/usr/local/mysql --datadir=/usr/local/mysql/data --plugin-dir=/usr/local/mysql/lib/plugin --log-error=/usr/local/mysql/data/mysqld.local.err --pid-file=/usr/local/mysql/data/mysqld.pid --keyring-file-data=/usr/local/mysql/keyring/keyring --early-plugin-load=keyring_file=keyring_file.so
nijia      89855  0.0  0.0  4296840    724 s000 R+    1:08下午  0:00.00 grep --color=auto --exclude-dir=.bzr --exclude-dir=CVS --exclude-dir=.git --exclude-dir=.hg --exclude-dir=.svn mysql
~▶ █
```





每次要找服務都要打一次

累得跟人一樣



35424504

since 2002/01/01

新手建議

開始閱讀之前

網站導覽

Linux 基礎文件

Linux 基礎訓練

Linux 架站文件

Linux 企業應用

Linux 安全管理

Linux 桌面應用

鳥哥彙整的資料

關於鳥哥

網友分享

特殊問題解決

網站資料搜尋



今日 0:1795
 昨日 04564
 本月 09802 :
 上月 29552 :

第十五章、例行性工作排程(crontab)

最近更新日期：2015/07/31

學習了基礎篇也一陣子了，你會發現到為什麼系統常常會主動的進行一些任務？這些任務到底是誰在設定工作的？如果你想要讓自己設計的備份程式可以自動的在系統底下執行，而不需要手動來啟動他，又該如何處置？這些例行的工作可能又分為『單一』工作與『循環』工作，在系統內又是哪些服務在負責？還有還有，如果你想要每年在老婆的生日前一天就發出一封信件提醒自己不要忘記，可以辦的到嗎？嘿嘿！這些種種要如何處理，就看看這一章先！

15.1 什麼是例行性工作排程

15.1.1 Linux 工作排程的種類：[at, crontab](#)

15.1.2 CentOS Linux 系統上常見的例行性工作

15.2 僅執行一次的工作排程

15.2.1 atd 的啟動與 at 運作的方式：[/etc/at.deny](#)

15.2.2 實際運作單一工作排程：[at, atq & atrm, batch](#)

15.3 循環執行的例行性工作排程

15.3.1 使用者的設定：[/etc/cron.deny, crontab](#)

15.3.2 系統的設定檔：[/etc/crontab, /etc/cron.d/*](#)

15.3.3 一些注意事項

15.4 可喚醒停機期間的工作任務

15.4.1 什麼是 anacron

15.4.2 anacron 與 [/etc/anacrontab](#)

15.5 重點回顧

15.6 本章習題

針對本文的建議：<http://phorum.vbird.org/viewtopic.php?t=23889>

15.1 什麼是例行性工作排程

但是今天不會教你用排程

Line Notify

HTTP Method

OPTIONS

GET

HEAD

POST

PUT

DELETE

TRACE

CONNECT

OPTIONS

GET

HEAD

POST

PUT

DELETE

TRACE

CONNECT

GET - 從指定的地方(IP)取得資源

POST - 紿指定的地方(IP)提交待處理資源

透過LINE接收其他網站服務通知

與網站服務連動完成後，LINE所提供的官方帳號「LINE Notify」將會傳送通知。

不僅可與多個服務連動，也可透過LINE群組接收通知。





以下為範例

從Mackerel接收伺服器的通知

與Mackerel連動 >



IFTTT在快下雨前傳送通知

與IFTTT連動 >



傳送GitHub的Event通知

與GitHub連動 >



簡單來說就是要做自動化通知

https://notify-bot.line.me/ja_TW/



Connect LINE with Everything

透過LINE接收其他網站服務通知

與網站服務連動完成後，LINE所提供的官方帳號「LINE Notify」將會傳送通知。

不僅可與多個服務連動，也可透過LINE群組接收通知。

Jia Yu Lin ▾

個人頁面

管理登錄服務

登出

管理登錄服務 (服務提供者用)

此為已登錄的服務一覽。

只要與LINE Notify連動，LINE用戶就能接收來自您所提供的網路服務通知。



埔里基督教醫院

Client ID o4zbLdahu4as4OUbVqqTJD

Users 0



台中市空氣品質

Client ID glGcF8FnVa0KBVkBNDfwm

Users 1



NiJia Notify

Client ID Eg9C7Nslhxmi6dhnyENJqT

Users 1

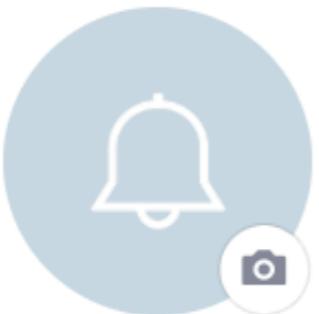


登錄服務

登錄服務

輸入以下各項目並完成電子郵件認證後，才能與LINE Notify連動。

服務圖片



服務名稱 •

服務概要 •

服務網址 •

企業／經營者名稱 •

所在國家／地區 •

選擇所在國家／地區

負責人姓名 •

電子郵件帳號 •

Callback URL •

※Callback網址最多可登錄5個。請以換行區隔不同的網址。

• 必填

[服務條款](#) [隱私權政策](#)

請於同意上述內容後再進行申請。

服務圖片



服務名稱 •

提醒者一號

服務概要 •

就是拿來做提醒的啦

服務網址 •

http://localhost:8080

企業／經營者名稱 •

NiJia有限公司

所在國家／地區 •

台灣

負責人姓名 •

NiJia

電子郵件帳號 •

louis70109@gmail.com

Callback URL •

http://localhost:8080

※Callback網址最多可登錄5個。請以換行區隔不同的網址。

• 必填

[服務條款](#) [隱私權政策](#)

請於同意上述內容後再進行申請。

同意並前往下一步

如果只要自己用就是用 localhost 就可以了

提醒者一號

輸入以下各項目並完成電子郵件認證後，才能與LINE Notify連動。

服務圖片



服務名稱

提醒者一號

服務概要

就是拿來做提醒的啦

服務網址

<http://localhost:8080>

企業／經營者名稱

NiJia有限公司

所在國家／地區

台灣

負責人姓名

NiJia

電子郵件帳號

louis70109@gmail.com

Callback URL

<http://localhost:8080>

編輯

登錄

LINE Notify註冊用網址通知

收件匣 



LINE Notify <do_not_reply@line.me>

寄給我 ▾

LINE Notify註冊用網址通知

感謝您註冊LINE Notify。

請前往註冊用網址，輸入您的個人資料及公司 / 經營者資訊。

【LINE Notify註冊用網址】

<https://notify-bot.line.me/my/services/complete?emailConfirmToken=BSyLCPqSFEIqyxblTUkoXj4M3tHGJOfgFCpGrEXd39d>

*本網址在郵件送出後24小時內有效。

【聯絡我們】

若您對申請內容有任何疑問，請由此與我們聯繫。

<https://contact.line.me/11672>

提醒您，本郵件為LINE Notify系統自動傳送，

請勿直接回信，謝謝。

<禁止轉載、轉寄、複製>

© LINE Corporation

← 回覆

→ 轉寄

管理登錄服務（服務提供者用）

此為已登錄的服務一覽。

只要與LINE Notify連動，LINE用戶就能接收來自您所提供的網路服務通知。



提醒者一號

Client ID CT18vBH0ijgpGsKNDSWh4D

Users 0



埔里基督教醫院

Client ID o4zbLdahu4as4OUbVqqTjD

Users 0



台中市空氣品質

Client ID glGcF8FnVa0KBVkBNDfwm

Users 1



NiJia Notify

Client ID Eg9C7Nslbymi6dhnyEN_IoT



提醒者一號

Client ID **CT18vBH0ijgpGsKNDSWh4D**

Client Secret **rNVMjjDR33qMZFtWYj8qsRh4y4Z22JmBVQd0P2wwAQz** 顯示 重設

Users **0**

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Notify</title>
5      <script>
6      function oAuth2() {
7          var URL = 'https://notify-bot.line.me/oauth/authorize?';
8          URL += 'response_type=code';
9          URL += '&client_id=[YOUR_CLIENT_ID]';
10         URL += '&redirect_uri=http://localhost:8080';
11         URL += '&scope=notify';
12         URL += '&state=N0_STATE';
13         window.location.href = URL;
14     }
15 </script>
16 </head>
17 <body>
18     <button onclick="oAuth2();">連結到 LineNotify 按鈕 </button>
19 </body>
20 </html>
```



提醒者一號

NiJia有限公司

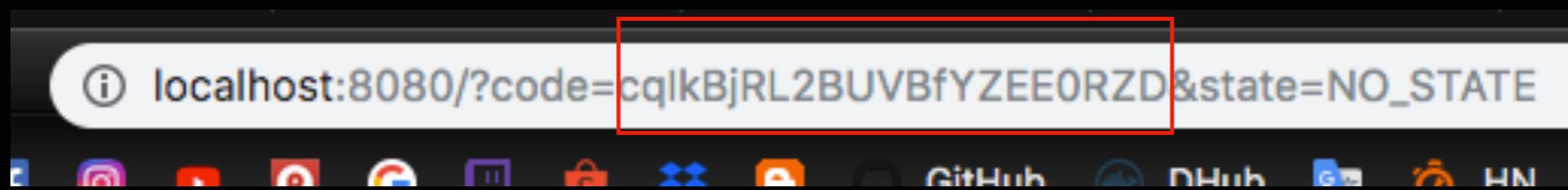
請選擇您要接收通知的聊天室。

Q Search by group name

- 透過1對1聊天接收LINE Notify的通知
- 20190628 Chatbots #10 @天壠書局
- BALLERS 球咖
- Chatbots Meetup - 聊天機器人小小聚
- Chatbots Meetup Core

個人資料上的姓名及已連動的聊天室名稱，將會提供給所連動服務的提供者。您可於LINE Notify的個人頁面解除連動。

同意後便會自動將「LINE Notify」官方帳號加入好友。



此時再開一個分頁

輸入這個網址並下載

<https://www.getpostman.com/downloads/>

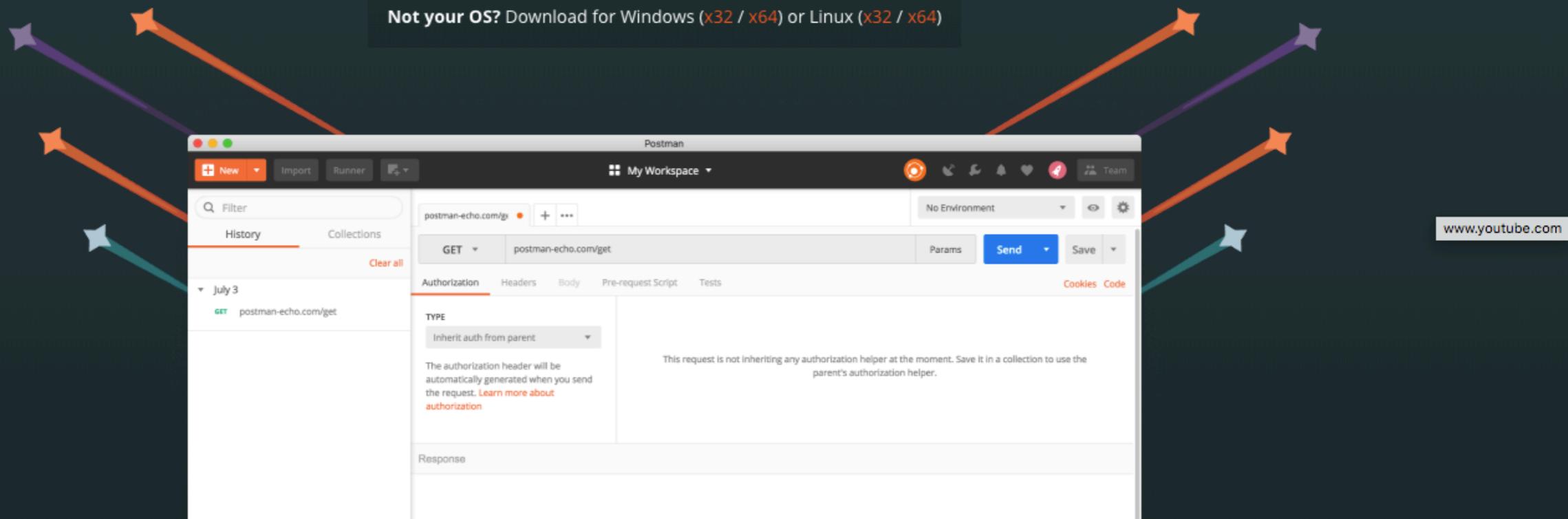
Get Postman for Mac

Join 6 million developers and download the **ONLY** complete API Development Environment.

 Download

Version 7.0.9 | [Release Notes](#)

Not your OS? Download for Windows (x32 / x64) or Linux (x32 / x64)



ries to ensure you get the
website. [Learn more](#)

ot it!

<https://notify-bot.line.me/oauth/token>

POST <https://notify-bot.line.me/oauth/token>

Params Authorization Headers **Body** ● Pre-request Script Tests

none form-data x-www-form-urlencoded raw binary

KEY	VALUE	DESCR
<input checked="" type="checkbox"/> grant_type	authorization_code	
<input checked="" type="checkbox"/> code	e0sOXjPAd7sNUUYmGFZupL	
<input checked="" type="checkbox"/> redirect_uri	http://localhost:8080	
<input checked="" type="checkbox"/> client_id	CT18vBH0ijgpGsKNDSWh4D	
<input checked="" type="checkbox"/> client_secret	rNVMjjDR33qMZFtWYj8qsRh4y4Z22JmBVQd0P2wwAQz	
Key	Value	Descr

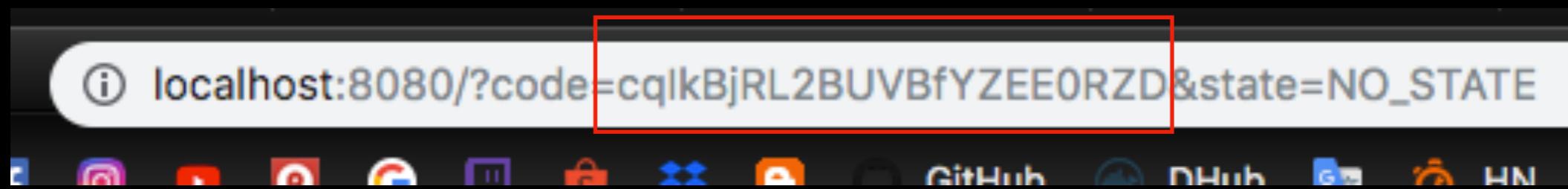
Body Cookies (1) Headers (13) Test Results

Pretty Raw Preview

JSON ▾



```
1 {  
2   "status": 200,  
3   "message": "access_token is issued",  
4   "access_token": "SSGcEFGNaZp4QA912ACTHXmgT4rbeTovUFNYeLbsc5t"  
5 }
```



<https://notify-api.line.me/api/notify>

POST

▼ <https://notify-api.line.me/api/notify>

Params

Authorization

Headers (1)

Body

Pre-request Script

Tests

KEY

VALUE

Authorization

Bearer SSGcEFGNaZp4QA912ACTHXmgT4rbeTovUFNYeLbsc5t

Key

Value

<https://notify-api.line.me/api/notify>

POST

<https://notify-api.line.me/api/notify>

Params

Authorization

Headers (1)

Body

Pre-request Script

Tests

none

form-data

x-www-form-urlencoded

raw

binary

KEY

VALUE



message

123456

Key

Value

Body

Cookies

Headers (11)

Test Results

Pretty

Raw

Preview

JSON



1 {

```
2     "status": 200,  
3     "message": "ok"  
4 }
```



到了這一步基本功能都已經完成了

The screenshot shows the Postman application interface. On the left, there's a sidebar with a 'History' section listing various API requests made today and in previous months. The main workspace shows a POST request to `https://notify-api.line.me/api/notify`. The 'Body' tab is active, displaying a JSON object with a single key-value pair: `"message": "你好..."`. A red arrow points from the 'Cookies' tab in the top right corner to the 'Code' tab in the bottom right corner of the body panel.

runner

My Workspace ▾

Invite

GENERATE CODE SNIPPETS

POST https://notify-api.line.me/api/notify

cURL

Copy to Clipboard

```
curl -X POST \
  https://notify-api.line.me/api/notify \
  -H 'Accept: */*' \
  -H 'Authorization: Bearer SSGcEFGNaZp4QA912ACTHXmgT4rbeTovUFNYeLbsc5t' \
  -H 'Cache-Control: no-cache' \
  -H 'Connection: keep-alive' \
  -H 'Host: notify-api.line.me' \
  -H 'Postman-Token: 6497d1b5-fc78-4a79-8d69-2e2604269c95,7e351864-ab3a-4320-bab6
-e79bc408dbff' \
  -H 'User-Agent: PostmanRuntime/7.11.0' \
  -H 'accept-encoding: gzip, deflate' \
  -H 'cache-control: no-cache' \
  -H 'content-length: 169' \
  -H 'content-type: multipart/form-data; boundary
-----WebKitFormBoundary7MA4YWxkTrZu0gW' \
  -F 'message=你好'
```

runner

My Workspace ▾

Invite

GENERATE CODE SNIPPETS

POST https://notify-api.line.me/api/notify

cURL

Copy to Clipboard

```
curl -X POST \
  https://notify-api.line.me/api/notify \
-H 'Accept: */*' \
-H 'Authorization: Bearer SSGcEFGNaZp4QA912ACTHXmgT4rbeTovUFNYeLbsc5t' \
-H 'Cache-Control: no-cache' \
-H 'Connection: keep-alive' \
-H 'Host: notify-api.line.me' \
-H 'Postman-Token: 6497d1b5-fc78-4a79-8d69-2e2604269c95,7e351864-ab3a-4320-bab6
-e79bc408dbff' \
-H 'User-Agent: PostmanRuntime/7.11.0' \
-H 'accept-encoding: gzip, deflate' \
-H 'cache-control: no-cache' \
-H 'content-length: 169' \
-H 'content-type: multipart/form-data; boundary=-----WebKitFormBoundary7MA4YWXkTrZu0gW' \
-F 'message=你好'
```

vim notify.sh

```
1  #!/bin/bash
2  echo "SHELL NAME: $0"
3  while true
4  do
5      read -r data
6      if [ "$data" = "exit" ]
7      then
8          echo "Bye"
9          exit 0
10     fi
11     echo "your input: $data"
12     curl -X POST \
13         https://notify-api.line.me/api/notify \
14         -H 'Authorization: Bearer SSGcEFGNaZp4QA912ACTHXmgT4rbetovUFNYeLbsc5t' \
15         -H 'Host: notify-api.line.me' \
16         -H 'content-type: multipart/form-data' \
17         -F "message=$data"
18     echo '\n-----\n'
19 done
```

sh notify.sh

做到這你已經完成了基本的任務通知程式

<https://ubin.io/tS2QaA>

接著需要的做的事就是弄成自動化

可以參考以下兩個

Crontab - <https://ppt.cc/fIORPx>

Supervisor - <https://ppt.cc/fjPzsX>

LINE Notify 後台導覽

Notify API 文件導覽

Notify 註冊與使用

串接第三方服務

空氣品質: <https://ppt.cc/f19xxx>

Twitch API: <https://ubin.io/5XjRAn>

我們寫的程式可以放在

<https://github.com/>

<https://bitbucket.org>

Q&A