

國立高雄科技大學

智慧商務系

程式設計期末報告

## Line Bot 提醒小工具



組員：

C112156210 陳韋均

C112156219 高宇希

C112156230 張庭濤

C112156242 陳筠樺

C112156244 許庭瑄

指導老師：謝文川 老師

中華民國一一四年一月

# 目錄

目錄 .....	I
圖目錄 .....	I
摘要 .....	1
第壹章、緒論 .....	2
一、前言 .....	2
二、研究動機 .....	2
三、研究目的 .....	2
四、研究目標 .....	3
五、研究流程 .....	4
第貳章、系統設計與架構 .....	5
一、開發工具 .....	5
二、系統設計 .....	5
三、功能規劃 .....	5
第貳章、預期成果 .....	7
第肆章、成果展示 .....	7
第伍章、可以改善的地方 .....	10
第陸章、結論及心得 .....	11
一、結論 .....	11
二、心得 .....	12
第柒章、參考資料 .....	14

# 圖目錄

圖 1 專題發展流程圖 .....	4
圖 2 系統架構圖 .....	5
圖 3 流程圖 .....	5

## 摘要

隨著數位化學習的迅速普及與深入發展，學生在學習過程中需要面對並同時處理大量的課程資訊、學術任務和課外活動，這種繁重而多樣化的需求使得傳統的手動記錄方式或依賴紙本課表來管理日程的方式已經逐漸顯現出其局限性，無法有效應對現代化學習環境中多變且高效的時間管理需求。

本研究旨在開發一個基於 Line Bot 的「課表提醒」機器人，為現代學生提供一個智能化、高效率的時間管理工具，解決在多課程、多任務的學習環境下，學生經常因資訊繁多而遺忘課程細節或重要任務。本系統結合 Line Bot 的便利性，讓學生能將 Line Bot 添加到群組聊天室，設置個人化的提醒架構，並在課程前一天收到自動提醒，此外，系統還支持設定子項目提醒、群組專案提醒，並提供未完成與已完成事項的查詢功能，進一步提升學生的時間管理能力與團隊協作效率。

功能特色包括自動爬取學生的課表資訊、根據課程安排提供精準提醒，以及透過簡單的指令設置和查詢提醒內容，學生可透過群組功能與同學協作，藉由提醒功能提升專案進度的透明度，確保各項任務順利完成。為確保系統的穩定性與安全性，我們採用了 SQLite 資料庫來保存提醒資訊，並規劃了穩定的伺服器部署與定期維護機制。

我們認為這套系統能夠幫助學生解決因繁忙課業造成的遺忘問題，這不僅提升學習效率，還能加強團隊溝通與協作能力，是一個兼具便利性與功能性的學習管理工具。

# 第壹章、緒論

## 一、前言

在現今的教育環境中，學生面臨著繁重的課業壓力和多樣化的學習任務，如何有效管理時間和任務成為了一個重要的課題。傳統的紙本課表和手動提醒方式已經無法滿足現代學生的需求，因此我們提出了利用 Line Bot 技術來開發一個智能課表提醒系統。這個系統不僅能夠自動提醒學生課程安排，還能夠管理課程下的子項目和群組專案，為學生提供一個全方位的學習管理工具。

## 二、研究動機

隨著數位化學習的普及，學生在學習過程中需要處理大量的資訊和任務，這使得時間管理變得更加複雜。許多學生因為行程繁忙而遺忘重要事項，導致學業成績受到影響。為了解決這個問題，我們希望開發一個能夠自動提醒學生課表和任務的系統，幫助學生更好地管理時間和任務，減少因記憶疏忽帶來的負面影響，提升學習效率和團隊合作的效果。

## 三、研究目的

本研究旨在解決學生在多課程、多任務的學習環境下容易遺忘課表與課程細節準備的問題，並提升團隊在專案和報告中的協作效率。透過開發一個基於 Line Bot 的課表提醒系統，我們希望能夠幫助學生更好地管理時間和任務，減少因記憶疏忽帶來的負面影響，從而實現更有效率的學業表現和專案成果。

具體目的如下：

1. 解決學生因時間管理不善而導致的重大問題。
2. 幫助學生提升時間管理能力。
3. 提高團隊協作效率。
4. 實現更有效率的學業表現和專案成果。

#### 四、研究目標

為了達成上述研究目的，本研究設定了以下具體目標。這些目標旨在通過開發和實施一個智能課表提醒系統，來提升學生的學習效率和團隊合作能力。

具體目標如下：

1. 開發一個基於 Line Bot 的課表提醒系統。
2. 實現學生輸入學號後自動爬取學校網站的課表資訊並回傳給學生。
3. 根據課表資訊在課程前一天自動提醒學生課程安排和教室位置。
4. 支持學生設置課程子項目提醒，並在前一天自動通知。
5. 提供群組專案提醒功能，支持@功能，並能夠查詢未完成和已完成的事項。

## 五、研究流程

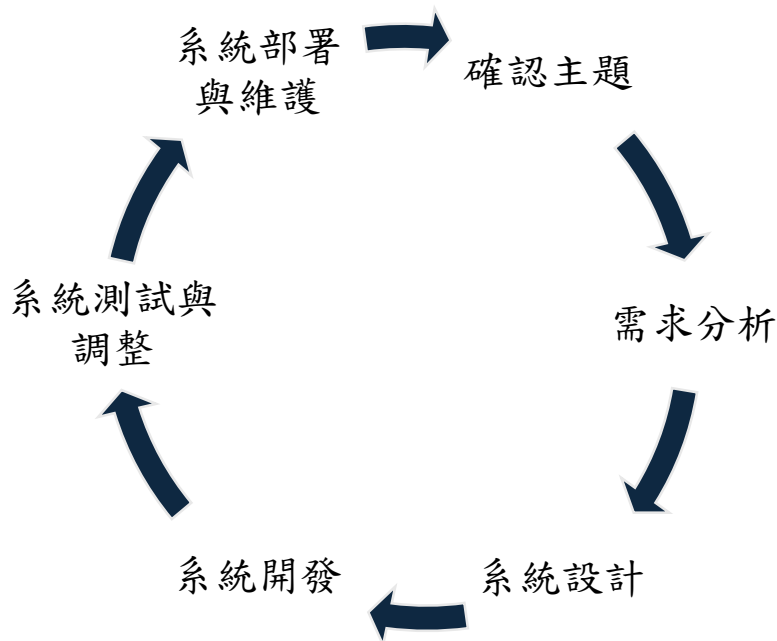


圖 1 專題發展流程圖

1. 確認主題：確立欲進行開發的主題為「課表及專案提醒機器人」。
2. 需求分析：確定系統需要實現的功能和特性，包括課表提醒、子項目提醒、群組專案提醒等，並進行可行性研究。
3. 系統設計：設計 Line Bot 的互動邏輯、提醒格式和使用界面。
4. 系統開發：實現系統功能，包括使用者輸入學號和密碼後爬取課表資訊、群組專案提醒功能等。
5. 系統測試與調整：對系統進行全面測試，確保系統功能正常運行，並修復發現的問題。根據使用者反饋進行調整和優化。
6. 系統部署與維護：部署伺服器，確保系統穩定運行，並定期進行維護和更新。

## 第貳章、系統設計與架構

### 一、開發工具

- 程式語言： Python
- 框架及套件： Flask、requests、bs4、linebot、apscheduler...
- 資料庫： Google sheet
- 編輯器： Visual Studio Code
- Line Bot API： 使用 Line 官方提供的 API 開發和管理 Bot 功能

### 二、系統設計

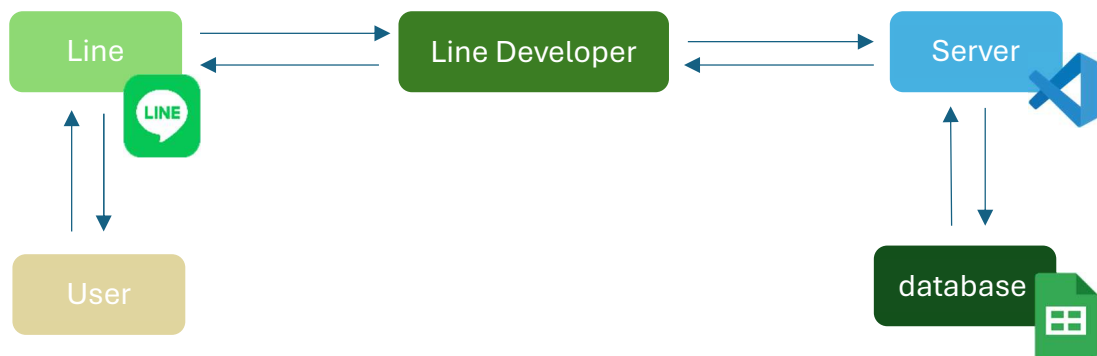


圖 2 系統架構圖



圖 3 流程圖

### 三、功能規劃

#### 1. 新增提醒

使用者在聊天室輸入指定指令"新增提醒"，機器人將回覆新增提醒的格式，依照格式輸入提醒內容，完成後系統會回覆確認訊息。

#### 2. 刪除提醒

使用者在聊天室輸入指令"刪除提醒"，系統將列出當前的所有提醒事項，使用者選擇欲刪除的項目即可。

### **3. 查詢提醒**

- **查詢未完成事項：**

使用者在聊天室輸入指令"未完成"，系統將回傳所有狀態為未完成之提醒事項。

- **查詢已完成事項：**

使用者在聊天室輸入指令"已完成"，系統將回傳所有已完成的提醒事項。

### **4. 定時提醒**

- **重複提醒：**

使用者在聊天室輸入指令"定時提醒"，即可依輸入時間重複推播提醒訊息。

- **每日提醒：**

機器人每日會在固定時間推播尚未完成的事項，提醒使用者盡速完成。

### **5. 課表查詢**

使用者在聊天室輸入指令"學號"及"密碼"後，系統將自動爬取課表並回傳目前學期的課程表。

### **6. 群組協作**

使用者加入群組後，系統將會自動傳送歡迎訊息及簡易操作說明。使用者可以在群組內執行以上所有功能。



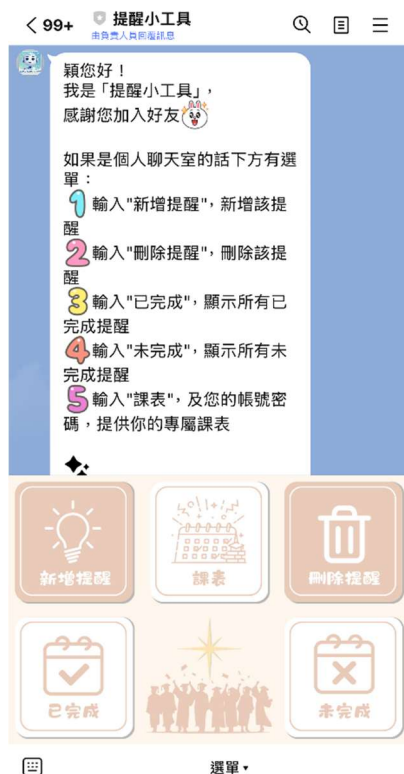
## 第貳章、預期成果

1. 完成 Line Bot 具備自動課表提醒功能的開發。
2. 透過 Line Bot 的課程提醒，增強學生的時間管理能力，減少課程及課堂重要事項的疏漏。
3. 透過 Line Bot 群組的專案任務提醒功能，提升團隊協作效率，促進溝通和進度管理。
4. 建立穩定的伺服器與資料庫系統，確保數據儲存和處理的安全性。
5. 建立功能完善的 LINE Bot，能夠實現新增、查詢、修改、刪除提醒等功能。
6. 擁有友好的使用者介面，讓使用者能夠輕鬆、前顯易懂得與 LINE Bot 互動。
7. 設定穩定可靠的定時提醒功能，能夠準確地在設定的時間發送提醒訊息。

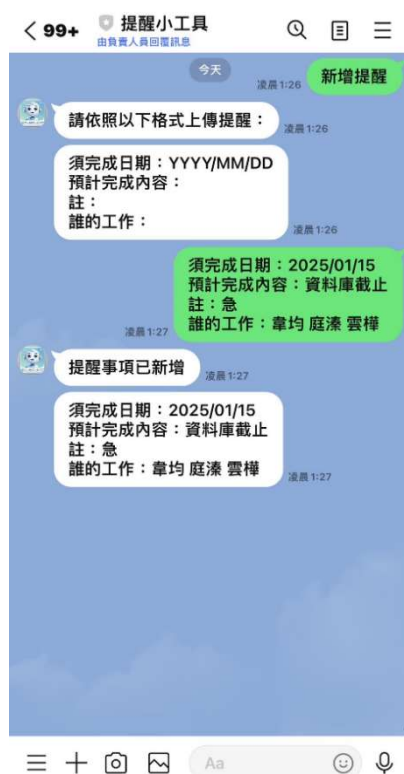
## 第肆章、成果展示



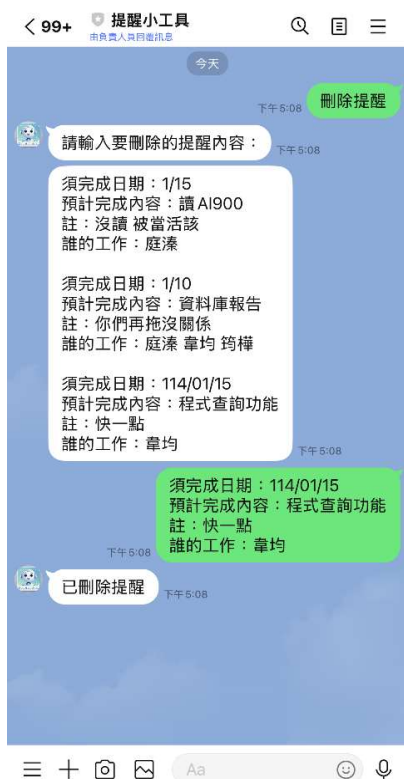
## 選單畫面



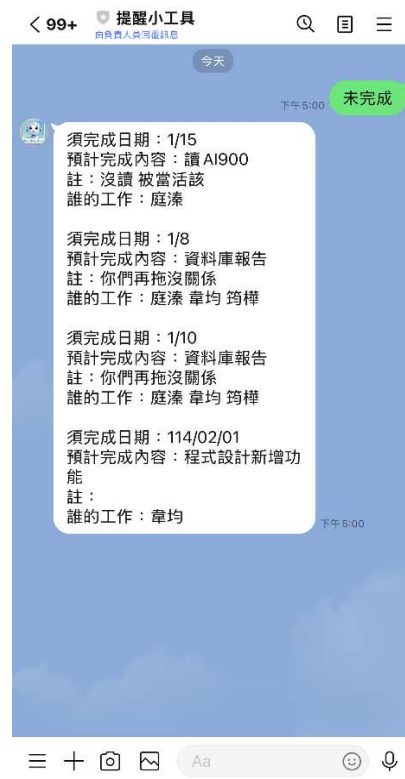
## 新增提醒



## 刪除提醒



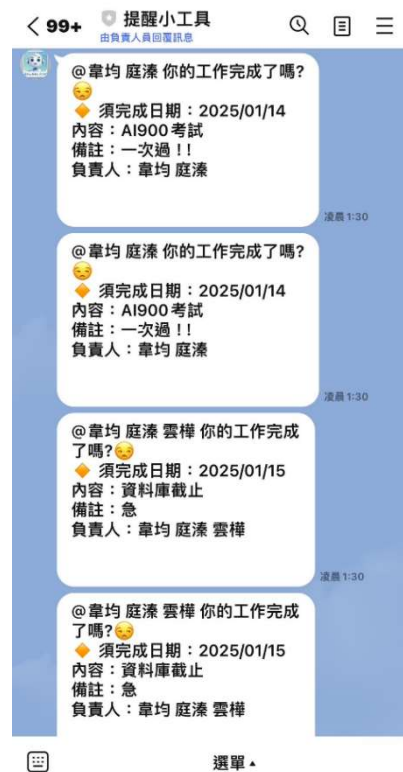
## 未完成的提醒



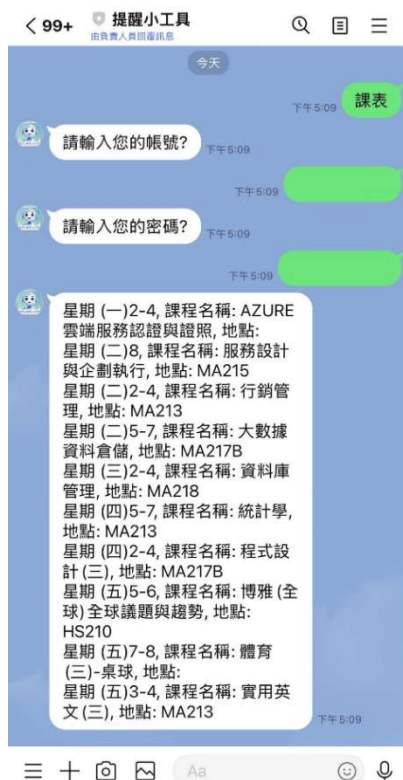
## 重複提醒



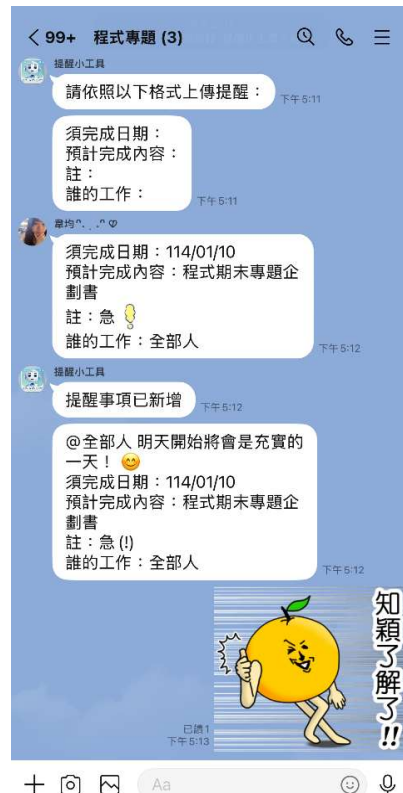
## 每日提醒



## 課表查詢



## 群組協作



## 第伍章、可以改善的地方

### 1. 功能延伸

- **Line bot 回覆訊息優化：**

我們可以將 label 功能運用在 Line bot 的訊息回覆上。例如，可以使用 label 回傳提醒並有按鈕可以直接標示已完成/未完成，來替代使用者打字的不便，增強用戶體驗。

- **語音指令：**

結合 Azure 的語音部分，增加語音輸入功能，用戶可以透過語音新增提醒或查詢事項，尤其適合開車或忙碌時的使用情境。

- **引入子任務提醒功能：**

因時間關係沒有設計子任務提醒（如：在處理課堂任務時，除了提醒課程時間和地點外，能夠額外提示用戶需要攜帶的工具或完成的準備工作），這確實是我們感到比較可惜的地方。子任務提醒可以進一步細化用戶的時間與任務管理，使整體系統更加完善，這將顯著提升提醒的實用性與精確性。

### 2. 技術改進

- **資料庫升級：**

目前使用 Google Sheet 進行資料存儲，可改用 SQLite 或 MySQL 等資料庫工具，以應對數據增長的需求。

- **API 使用最佳化：**

利用非同步操作（如 Asyncio）提升 API 的呼叫效率，避免高

流量時的延遲問題。

### 3. 使用者體驗：

- **可不按照格式，機器人會自己判斷關鍵字並回覆：**

此設計將會大大降低操作複雜性，使用者不再需要記住特定格式，只需輸入關鍵字即可與系統互動，讓操作更加簡單直觀，即使輸入的格式不完全正確，系統也能準確理解並提供有效回應，減少操作失敗的情況。此外機器人根據語意或關鍵字智能回覆，也能讓使用者感覺更自然，像與人對話一般。

## 第陸章、結論及心得

### 一、結論

這個專案開發了一個基於 Line Bot 的課表及提醒管理系統，成功整合爬蟲技術、自動化提醒功能和群組協作工具，實現了多項功能目標。

核心功能包括：

1. **課表查詢：**透過爬蟲自動提取課表資訊，並將結果以簡潔明瞭的格式回傳給用戶。
2. **提醒管理：**支援新增、刪除、查詢未完成及已完成的提醒，滿足學生的日程與任務管理需求。
3. **群組協作：**在群組中提供專案任務的提醒與進度查詢，提升團隊協作效率。

4. **定時與每日提醒**：利用排程技術實現自動化提醒，確保用戶不會遺漏重要事項。

成就與影響：

- 幫助學生在繁忙的課業中有效管理時間與任務，減少因遺忘造成的負面影響。
- 強化團隊專案管理與溝通透明度，促進學術與合作成果的提升。

此專案成功地運用現代工具（如 Flask、Google Sheets API 和 LINE Messaging API），並結合多方技術達成系統的穩定性和實用性，無論是 API 集成、數據管理還是使用者訊息通知，都得以高效且穩定地完成，充分體現了現代學生課業繁忙的問題，也滿足了學生的需求。此外，這些技術還能夠根據未來需求進行靈活擴展和優化，使系統在長期運行中保持優勢，適應更多的應用場景和用戶需求。

## 二、心得

C112156210 陳韋均

在這次的專題開發過程中，我們的目標是開發一個基於 Line Bot 的課表提醒系統，從功能設計到實際開發，再到最後的測試與調整，每一個環節都充滿了學習與成長的機會。

首先，我們嘗試將「大數據倉儲」課程所學到的爬蟲技術應用於系統開發中，透過爬取校務系統的課表資料，將這些資訊整合到 Line Bot 中，並將其轉換為使用者能夠直觀理解的格式。再來我們又將提醒訊息以 CRUD 的方式呈現，並支援將機器人加入群組，以便在專案討論時，可以更直觀、清楚的讓組員了解代辦事項。

在專案開發中，我們也學到了如何整合多個 API，如 Line Messaging API 和 Google Sheets API，實現了提醒資料的即時存取與執行。對於初學者來說，這些技術的學習並不輕鬆，但透過一次次的測試與修正，我們逐漸掌握了這些工具的使用方法，也對程式設計的邏輯性與實用性有了更深刻的認識。

這次的專案開發讓我不僅收穫了技術上的進步，更對程式設計有了全新的理解，相信這次的經驗將會成為我未來學習與工作的堅實基礎，也讓我對未來的程式開發充滿期待。

#### C112156219 高宇希

這次的程式專題是我第一次嘗試使用爬蟲來製作主題。過程中，雖然有時會因為卡住或程式錯誤而感到挫折，但我的組員們都非常優秀且分工明確，讓整個專案得以順利進行。在每次遇到問題時，我的組員總是能夠即時的支援與解決問題。這次的專題讓我更了解如何使用爬蟲且增加了實務經驗，讓我對程式設計與爬蟲相關領域有更深入的了解。

#### C112156230 張庭溱

這次 Line Bot 的專題作業，對我來說是一個非常有挑戰但也很有收穫的過程，我們組的主題是提醒與查閱課表功能的 line bot，我負責設計圖文選單並撰寫部分程式碼，透過這次的專案，我們運用了課堂上學到的知識完成了 Line Bot 的功能建立，並且將大數據課程中學到的爬蟲技術應用在課表查詢功能上，雖然過程中遇到了不少挑戰，但靠著組員的合作與支持，我們也克服了不少問題，讓專案逐漸完善起來，即使最終成果可能不如其他組那麼出色，但我依然很滿意，因為這次經歷

讓我收穫了許多實用的技能，也增強了團隊合作的經驗。

C112156242 陳筠樺

這次的專題是我第一次製作機器人，在過程中，我對 Python 函式的運用有了更深入的理解，透過此次的實作，也使我快速檢視自己在這學期課程中的學習成果。

在專題中，我主要負責將 LINE Bot 與資料庫進行連結，從一開始完全不知道該如何下手，到最後能熟練運用雲端資料庫，這段學習過程讓我獲益良多，尤其是在學習 Google Cloud API 這項功能上，我克服了许多挑戰，最後成功完成了相關功能的開發，這讓我感到非常有成就感，也收穫了豐富的經驗，期待未來自己能將這項技能運用在更多實作上。

C112156244 許庭瑄

這次的報告對於我的組員我感到很抱歉，在過程中我因為私人原因沒有付出多少努力，但在過程中，我的組員們依舊很努力的幫助我，在我不會的地方都盡全力的教我，透過這次的報告我學習並更了解到了有關 linebot 以及 github 上傳的方法。

## 第柒章、參考資料

[1] Line Bot API。

<https://hackmd.io/@xia20020219/rkjCEPWJY>

[2] 【Canva 教學】如何用 Canva 做出 LINE 圖文選單？

<https://socialgenie.shoper.vip/introduction-to-line-menus/>



[3] LINE BOT 教學

<https://steam.oxxostudio.tw/category/python/example/line-bot.html>

[4] DAY 7 回話機器人(鸚鵡 LINE Bot)與 ngrok。

<https://ithelp.ithome.com.tw/articles/10295654>