# 座位表

座位表 -請同學填入 自己名字(有看到的提醒隔壁的同學 填寫 感謝!!

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **白板** | | | | | |
| 楊凱元 | 陳瑀萱 | 翁紹捷⭐ | 吳承祐 | 吳晉宇 | 助教 |
| 戴中航 | 沈咨名 | 班代:翁尉珊 | 程韋鈞 | 趙冠宇 | 譚峻 |
| 連彥鈞 | 劉樾 | 許瑀芹 | 王俐斐 | 葉濠銘 | 張雲藝 |
| 張書榜 | 王祥鈞 | 張又心 | 許媁涵 | 方俊程 | 蔡定芝 |
| 😂😂😂 | | 蘇煜㭿 | 林苔英 | 簡緯翔 | 曾祥易 |
| 黃玟勝(+100) |  | 張竣豪 | 黨敏伶 |

# 

# 課程流程⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐

1. 雲原生是什麼
2. Colab
3. 教材準備
4. 教材體驗第一步-Flask
5. Line 開發者帳號申請
6. Line 功能逐步體驗

# <https://github.com/line/line-bot-sdk-python>

# 雲原生是什麼

以後大家的電腦都不需要再去裝任何程式語言，

未來的主流就是大家的電腦環境都一樣。

Google, Amazon

提供一個標準環境，無論是誰，什麼時候連入，環境都是一樣的，Python版本也都會是一樣。

Google出了Colab，開發軟體，滿足上面的需求，還想要滿足資料科學家，資料科學家就想要顯卡(GPU)。

準備到下一步環節，去使用Colab吧 ↓↓↓

<https://colab.research.google.com/>

# 

# Colab

# 

# 

# 教材準備

### 到此網址，下載所有教材程式碼

<https://github.com/BingHongLi/ai-chatbot-tutorial>

解壓縮後，上傳到自己的google drive內

# 

# 教材體驗第一步-Flask

## Summary

因為Line跟我們溝通的方式

用戶 → Line → 我方電腦

Line傳給我們的方式是網頁常用的協定，http

我們必須要讓我們的電腦變成http server

每一種程式語言都可以開發http server

我們用Python，最常聽到的方式就是使用flask (或者Django)

## 流程

用戶→連結Colab→執行http網址

用戶→(可直接透過瀏覽器)訪問網址 (使colab變成http server)

啟動001 程式碼

修改一小段內容

執行

觀看結果

001程式碼解釋

1.安裝flask、line-bot-sdk套件

!pip install

! = cmd command

2.引用套件

# 

### 必須新增的程式碼

'''

用戶訪問 這個app的路徑abc，會執行ai-demo這個方法

ai-demo這個方法會回傳 tibame這個字串給用戶

'''

# 啟動主程序的API接口，最簡單版本

@app.route("/abc", methods=['GET'])

def ai\_demo():

return 'tibame這個字串給用戶'

打完之後，停止執行colab，重新執行

### Summary

在人類的世界裡，會有很多的異性，為了要能夠談戀愛，我們必須要用一些共同的方式溝通，比如眼神，說話

在網際網路的世界裡，會有很多的網民消費者，為了要能夠有效跟他們溝通，最常用的是http server的方式

於是我們展示了python編寫http server的方式，並希望大家能打一遍，從中發現它的規律

#### 規律

1. 安裝套件
2. 引用套件
3. 主程序建置
4. 為主程序編寫入口，以及入口後面的方法
5. 主程序執行

### 衍伸提問

#### ?是什麼

query string(查詢字串)

給我們server使用的參數

?變數1=變數內容&變數2=變數內容

example: xxxxxx.ngrok.io/calculate?v1=50&v2=90

#### 註解

# 單行註解

'''''' 多行註解

“””””” 多行註解

很多同學會不小心把程式碼的關鍵語句給註解掉。

# Line 開發者帳號申請

### Line的溝通流程梳理

消費者 → 訂單 → 郵局 → 公司地址 → 公司大門 → 總機 → 總機把消息轉發處理 → 回信人員→ 郵局 → 消費者

Line用戶 → 消息 → LINE 平台 → Webhook(https網址) → app的route → Webhookhandler收信 → Webhookhandler處理並轉發 → LineBotApi 把消息傳回給 LINE → LINE用戶

### 看一下程式碼

1. 引用套件
2. 設定app
3. 建置line\_bot\_api, handler (用來handle Events，例如Followevent、Unfollowevent)
4. app.route() 方法，消息稽核過濾，把那些消息都寫在log檔裡面，handler轉發
5. 告知handler 如果收到關注事件(FollowEvent)，就做下面的方法
   1. 取個資，存起來
   2. 回話給用戶
6. 運行app

帳號沒辦，就沒有channel\_access\_token, channel secret

## 辦帳號啦

google 搜尋Line messaging api

點擊右上角Log in

<https://developers.line.biz/zh-hant/docs/messaging-api/overview/>

登入啦~透過行動條碼

# 

# 

# Line 功能逐步體驗

關注的就忘了它吧，我們忘記筆記了

用戶發消息的

第一件 消息接收與處理

handler.add(event)

event

FollowEvent

UnfollowEvent

MessageEvent

TextMessage

ImageMessage

etc

第二件事情 用戶發來的TextMessage

官方建議加上特殊符號

NLP

用戶傳來 安安

我們跟他說 幾歲住哪

規則型

用戶傳來 安安

我們的機器人到底要做 規則 還是NLP

規則 → @安安

NLP → 安安

第三件事情 TextMessage 與 TextSendMessage

TextMessage: 用戶傳來的

TextSendMessage: 我們傳給用戶的

# 範本訊息的動作

動作複製起來，貼回程式碼

{

"type": "postback",

"label": "企業，查找商業結合方案",

"text": "[::text:]尋找BD",

"data": "Data1"

},

{

"type": "postback",

"label": "開發者，尋求教學",

"text": "[::text:]求助專家",

"data": "Data2"

},

{

"type": "uri",

"label": "打電話給班代",

"uri": "tel://0936903210"

}

----- LINE URL Scheme+

{

"type": "postback",

"label": "企業，查找商業結合方案",

"text": "[::text:]尋找BD",

"data": "Data1"

},

{

"type": "postback",

"label": "開發者，尋求教學",

"text": "[::text:]求助專家",

"data": "Data2"

},

{

"type": "uri",

"label": "拍照",

"uri": "https://line.me/R/nv/camera/"

},

{

"type": "uri",

"label": "發地理消息",

"uri": "https://line.me/R/nv/location/"

}

URI動作的戰略價值

<https://developers.line.biz/en/docs/messaging-api/using-line-url-scheme/#available-line-url-schemes>



007:

範本訊息

-json-linebotdesigner

-按鍵-動作---1.訊息動作

2.uri動作--- a.導外部

b.拍照

c.定位

3.回傳動作:可以做if else判斷、特殊功能、貼標

009:

最終展示

1.關注，取個資

2.用戶發話，我們回應

3.用戶發話，觸發範本

4.用戶發話，觸發QuickReply

5.取得用戶照片

４.

範本

一直存在

不適用一次性場合

問答-->QuickReply

-->用戶發話，我們回應

-->文字圖片順帶去QuickReply

推播

同一個Provider底下取得的同用戶user\_id是一樣的

單體push

全體broadcast

精準推播narrowcast  
<https://engineering.linecorp.com/zh-hant/blog/narrowcast-api-on-line-chatbot/>

客製化推播multicast

# Troubleshooting