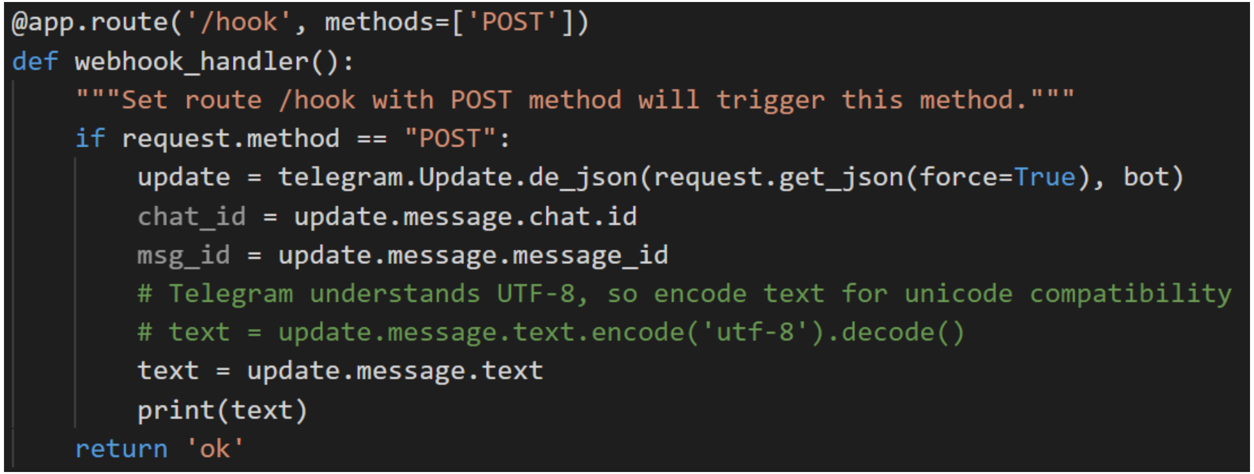
簡單訊息回應

安裝必要的lib

- Flask
  - https://flask.palletsprojects.com
  - pip install Flask
- python-telegram-bot
  - https://github.com/python-telegram-bot/python-telegram-bot
  - pip install python-telegram-bot --upgrade

使用 Flask 架設 web server

- 架設 web server 並且與 Telegram Bot 連結
  - Handle 所收到的 POST 請求

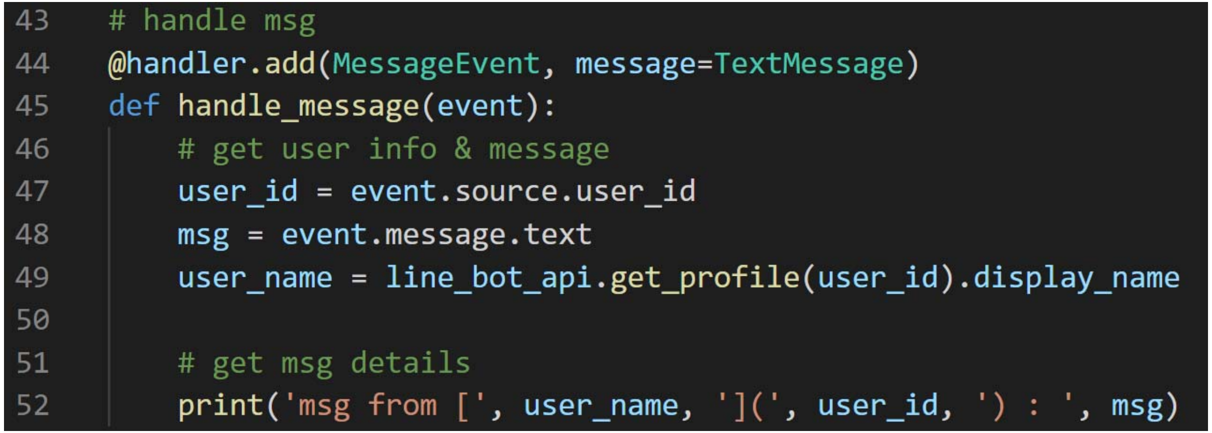


```python
@app.route('/hook', methods=['POST'])
def webhook_handler():
    """Set route /hook with POST method will trigger this method."""
    if request.method == "POST":
        update = telegram.Update.de_json(request.get_json(force=True), bot)
        chat_id = update.message.chat.id
        msg_id = update.message.message_id
        # Telegram understands UTF-8, so encode text for unicode compatibility
        # text = update.message.text.encode('utf-8').decode()
        text = update.message.text
        print(text)
    return 'ok'
```

- Telegram Platform 將用戶訊息轉換為 JSON

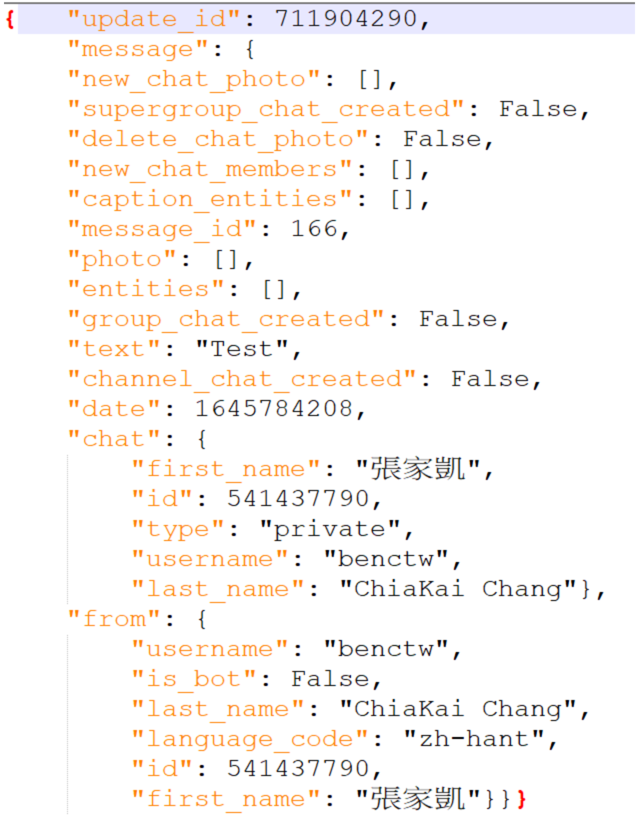
接收用戶訊息

- Telegram Platform 將用戶訊息轉換為 JSON 後
  - 可以分類為多種 Webhook event，最常見為用戶訊息事件 (MessageEvent)
  - 使用 Python decorator 搭配 handler.add 宣告欲接收並處理的 event
- 範例：



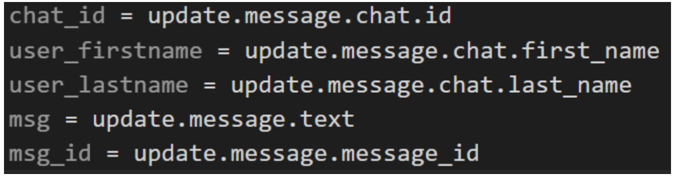
```python
# handle msg
@handler.add(MessageEvent, message=TextMessage)
def handle_message(event):
    # get user info & message
    user_id = event.source.user_id
    msg = event.message.text
    user_name = line_bot_api.get_profile(user_id).display_name

    # get msg details
    print('msg from [', user_name, '](', user_id, ') : ', msg)
```

接收用戶訊息



```
{       "update_id": 711904290,
        "message": {
        "new_chat_photo": [],
        "supergroup_chat_created": False,
        "delete_chat_photo": False,
        "new_chat_members": [],
        "caption_entities": [],
        "message_id": 166,
        "photo": [],
        "entities": [],
        "group_chat_created": False,
        "text": "Test",
        "channel_chat_created": False,
        "date": 1645784208,
        "chat": {
            "first_name": "張家凱",
            "id": 541437790,
            "type": "private",
            "username": "benctw",
            "last_name": "ChiaKai Chang"},
        "from": {
            "username": "benctw",
            "is_bot": False,
            "last_name": "ChiaKai Chang",
            "language_code": "zh-hant",
            "id": 541437790,
            "first_name": "張家凱"}}}
```

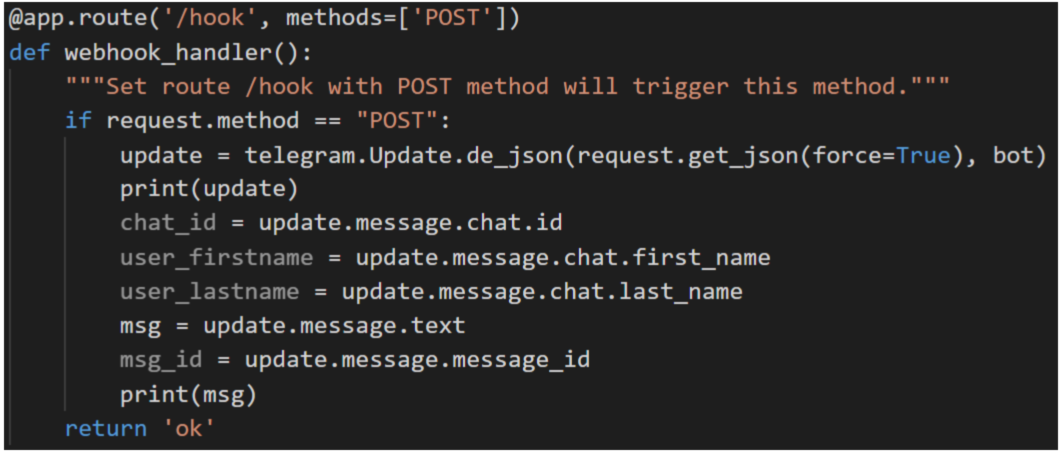
- 解析 event 內容，獲得用戶資訊與訊息
  - event 由 JSON 格式轉成
  - 參考 JSON 格式解析資料，取得用戶專屬ID與傳送之訊息
  - 利用 Python 取得用戶於 Telegram 上設定之姓名



```
chat_id = update.message.chat.id
user_firstname = update.message.chat.first_name
user_lastname = update.message.chat.last_name
msg = update.message.text
msg_id = update.message.message_id
```

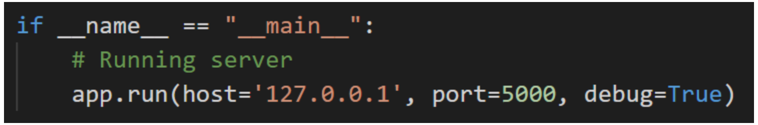
接收用戶訊息

- 將取得之資料印出



```
@app.route('/hook', methods=['POST'])
def webhook_handler():
    """Set route /hook with POST method will trigger this method."""
    if request.method == "POST":
        update = telegram.Update.de_json(request.get_json(force=True), bot)
        print(update)
        chat_id = update.message.chat.id
        user_firstname = update.message.chat.first_name
        user_lastname = update.message.chat.last_name
        msg = update.message.text
        msg_id = update.message.message_id
        print(msg)
    return 'ok'
```

- 呼叫 app.run() 啟用 web server



```
if __name__ == "__main__":
    # Running server
    app.run(host='127.0.0.1', port=5000, debug=True)
```

回覆用戶訊息

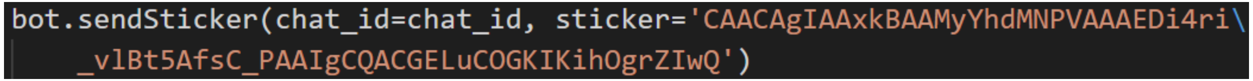
- 在 update 的 JSON 中，存有一個 chat_id，其實就是使用者獨一無二ID
  - 可使用於推送Telegram訊息 (主動推送訊息)
  - 也可用於即時回覆訊息 (被動回覆訊息)
  - 在Telegram中，發送訊息不論主動或被動式回覆，均不需要付錢
- Telegram發送用戶的訊息依照不同類型分為不同物件，如：
  - 文字訊息：sendMessage (SDK, API)
  - 貼圖： sendSticker (SDK, API)
  - 圖片訊息：sendPhoto (SDK, API)
  - 影片訊息：sendVideo (SDK, API)
  - 聲音訊息：sendAudio (SDK, API)
  - 地點：sendLocation (SDK, API)

回覆用戶訊息

- Telegram發送訊息
  - bot.sendMessage(chat_id=chat_id, text=text)
  - 也可以加上引用：
  - bot.sendMessage(chat_id=chat_id, text=text, reply_to_message_id=msg_id)

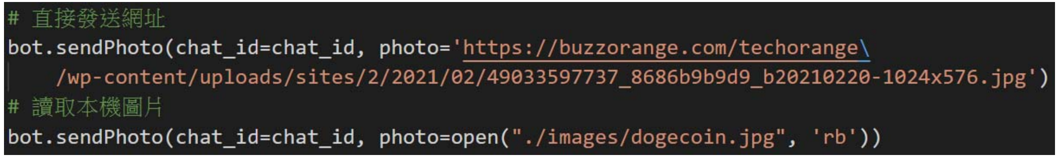
回覆用戶訊息

- 貼圖：



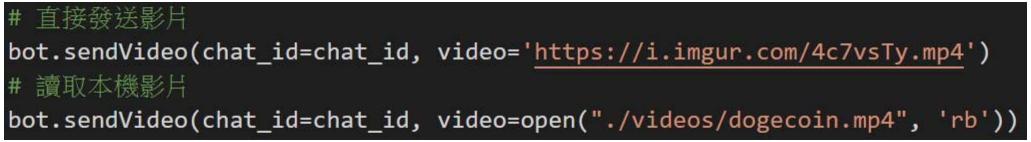
```
bot.sendSticker(chat_id=chat_id, sticker='CAACAgIAAxkBAAMyYhdMNPVAAAEDi4ri\
    _vlBt5AfsC_PAAIgCQACGELuCOGKIKihOgrZIwQ')
```

- 圖片：



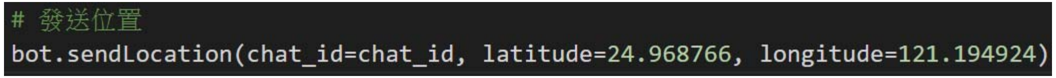
```
# 直接發送網址
bot.sendPhoto(chat_id=chat_id, photo='https://buzzorange.com/techorange\
    /wp-content/uploads/sites/2/2021/02/49033597737_8686b9b9d9_b20210220-1024x576.jpg')
# 讀取本機圖片
bot.sendPhoto(chat_id=chat_id, photo=open("./images/dogecoin.jpg", 'rb'))
```

- 影片：



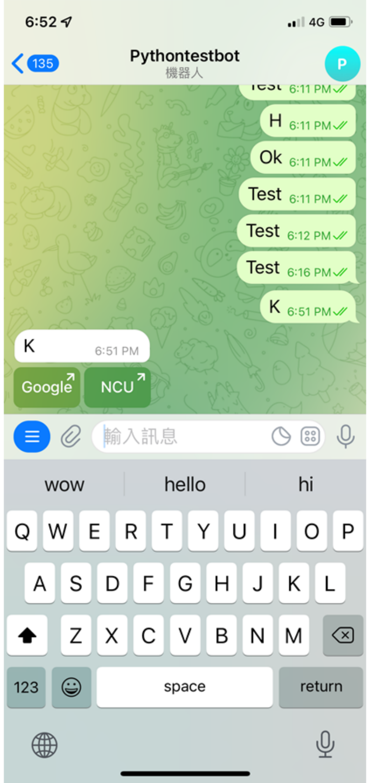
```
# 直接發送影片
bot.sendVideo(chat_id=chat_id, video='https://i.imgur.com/4c7vsTy.mp4')
# 讀取本機影片
bot.sendVideo(chat_id=chat_id, video=open("./videos/dogecoin.mp4", 'rb'))
```

- 地點：



```
# 發送位置
bot.sendLocation(chat_id=chat_id, latitude=24.968766, longitude=121.194924)
```

inline_keyboard


6:52

Pythontestbot
機器人

Test 6:11 PM

H 6:11 PM

Ok 6:11 PM

Test 6:11 PM

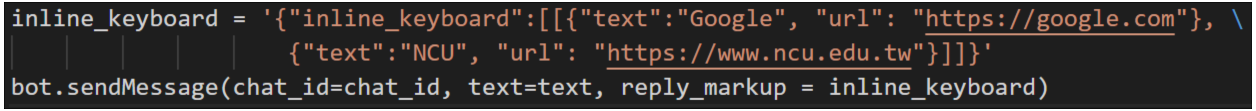
Test 6:12 PM

Test 6:16 PM

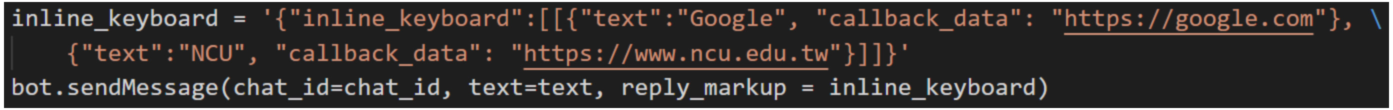
K 6:51 PM

K 6:51 PM

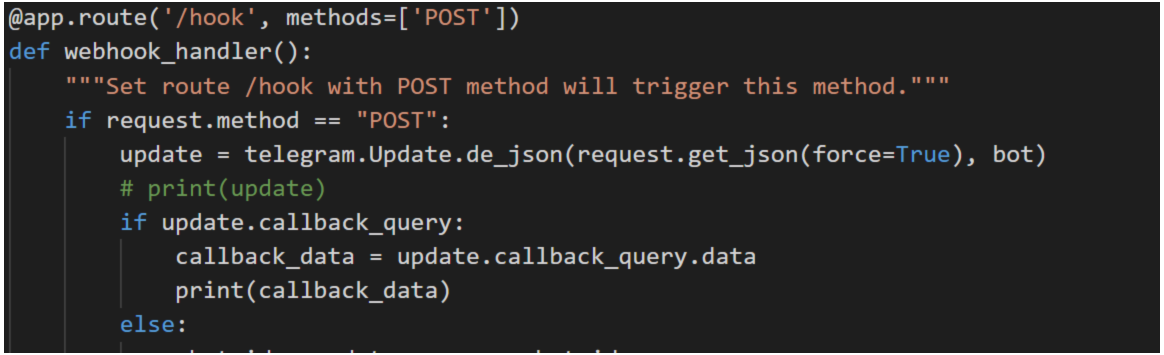
URL


```python
inline_keyboard = '{"inline_keyboard":[[{"text":"Google", "url": "https://google.com"}, \
                    {"text":"NCU", "url": "https://www.ncu.edu.tw"}]]}'
bot.sendMessage(chat_id=chat_id, text=text, reply_markup = inline_keyboard)
```

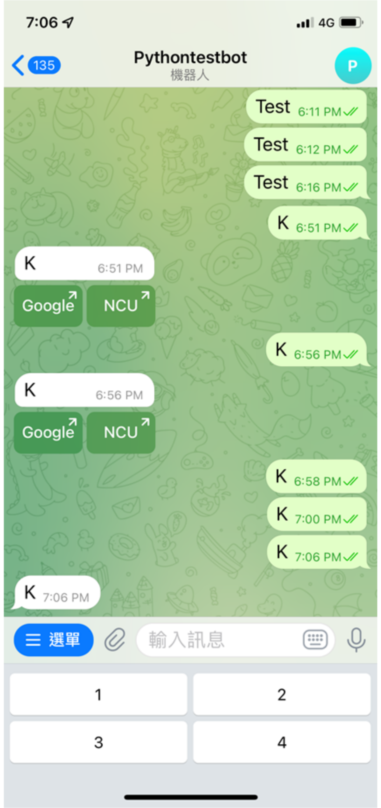
callback_data


```python
inline_keyboard = '{"inline_keyboard":[[{"text":"Google", "callback_data": "https://google.com"}, \
    {"text":"NCU", "callback_data": "https://www.ncu.edu.tw"}]]}'
bot.sendMessage(chat_id=chat_id, text=text, reply_markup = inline_keyboard)
```

讀取Telegram回傳callback_data訊息


```python
@app.route('/hook', methods=['POST'])
def webhook_handler():
    """Set route /hook with POST method will trigger this method."""
    if request.method == "POST":
        update = telegram.Update.de_json(request.get_json(force=True), bot)
        # print(update)
        if update.callback_query:
            callback_data = update.callback_query.data
            print(callback_data)
        else:
```

reply_keyboard



Test 6:11 PM

Test 6:12 PM

Test 6:16 PM

K 6:51 PM

K 6:51 PM

K 6:56 PM

K 6:56 PM

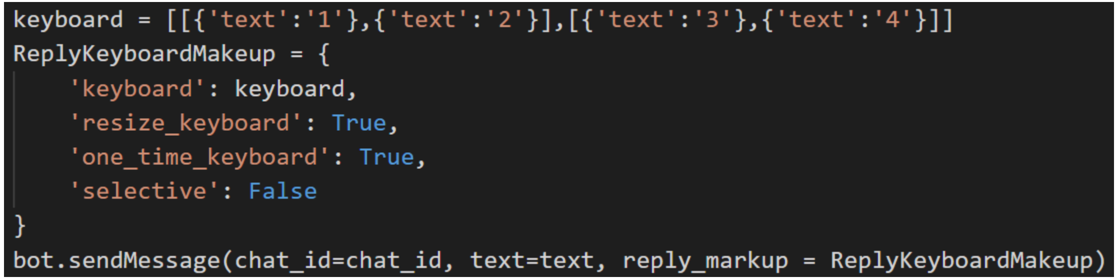
K 6:58 PM

K 7:00 PM

K 7:06 PM

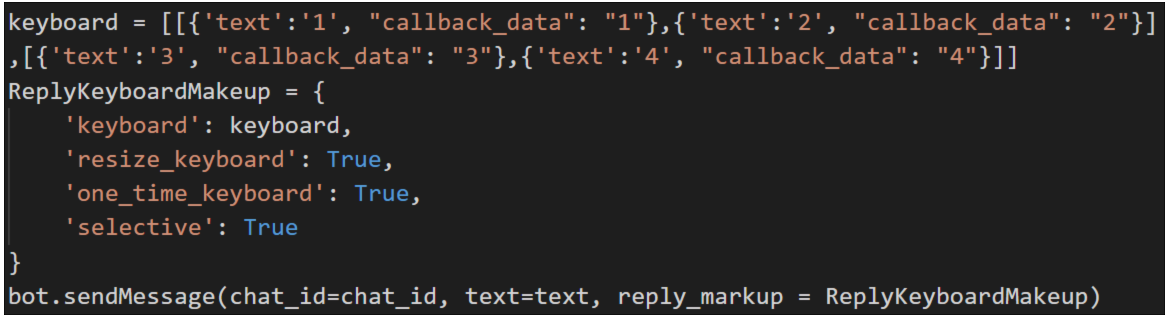
K 7:06 PM

■Text



```
keyboard = [[{'text':'1'},{'text':'2'}],[{'text':'3'},{'text':'4'}]]
ReplyKeyboardMakeup = {
    'keyboard': keyboard,
    'resize_keyboard': True,
    'one_time_keyboard': True,
    'selective': False
}
bot.sendMessage(chat_id=chat_id, text=text, reply_markup = ReplyKeyboardMakeup)
```

■Callback_data



```
keyboard = [[{'text':'1', "callback_data": "1"},{'text':'2', "callback_data": "2"}]
,[{'text':'3', "callback_data": "3"},{'text':'4', "callback_data": "4"}]]
ReplyKeyboardMakeup = {
    'keyboard': keyboard,
    'resize_keyboard': True,
    'one_time_keyboard': True,
    'selective': True
}
bot.sendMessage(chat_id=chat_id, text=text, reply_markup = ReplyKeyboardMakeup)
```