

## WeHelp

### Assignment - Week 6

---

延續上週任務建立的資料庫和資料表，整合前端、後端，完成會員註冊、登入、登出系統。在這次的作業中會建立 Python Flask 網站專案，支援前端頁面，並進行資料庫操作，可參考 [W3Schools Python MySQL](#) 的教學使用 MySQL 官方提供的 [mysql-connector-python](#) 套件連結資料庫。

**注意：**GitHub Repository 為公開場域，不要在程式中使用重要或常用的密碼。

#### 要求一：建立前端頁面

在這個會員系統中包含三個主要的頁面，請參考以下說明完成：

網站首頁：

- 網址：<http://127.0.0.1:3000/>
- 連線方法：GET
- 參考設計如下：密碼請使用密碼輸入框完成

### 歡迎光臨，請註冊登入系統

#### 註冊帳號

姓名

帳號

密碼

註冊

#### 登入系統

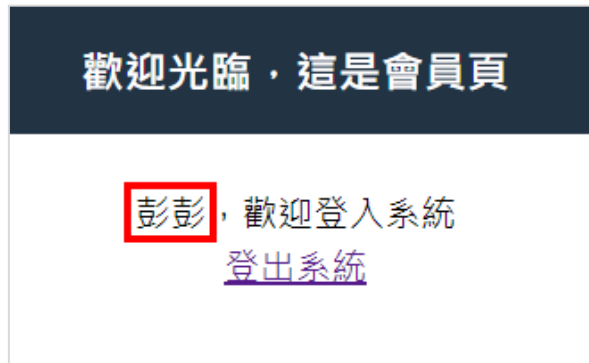
帳號

密碼

登入

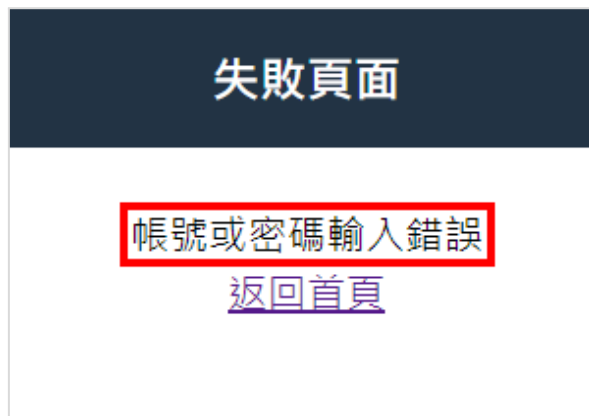
會員頁面：

- 網址：<http://127.0.0.1:3000/member>
- 連線方法：GET
- 參考設計如下：登入成功後的會員頁，要包含登入的會員姓名 (name)。



失敗頁面：

- 網址：<http://127.0.0.1:3000/error?message=自訂的錯誤訊息>
- 連線方法：GET
- 參考設計如下：錯誤訊息，必須從網址的 Query String 取得。



## WeHelp

### Assignment - Week 6

---

#### 要求二：建立註冊帳號程序

使用者第一次進入網站，需要在首頁註冊一個新的帳號。

註冊功能端點：

- 網址：<http://127.0.0.1:3000/signup>
- 連線方法：POST

必須滿足的運作流程：

1. 使用者在表單中，輸入姓名 (name)、帳號 (username)、密碼 (password)，點擊【註冊】按鈕。前端使用 JavaScript 檢查是否有未輸入的資訊，若有，阻止表單送出。
2. 連線到【註冊功能端點】，後端接收前端傳進來的資料，並按照以下程序處理：
  - a. 檢查資料庫的 member 資料表中是否有重複的帳號 (username)
    - i. 如果有重複，不新增資料，註冊失敗，導向到【失敗頁面】，並顯示「帳號已經被註冊」
    - ii. 如果沒有重複，新增資料到資料庫的 member 資料表，註冊成功，導向到【網站首頁】

## WeHelp

### Assignment - Week 6

---

#### 要求三：建立登入帳號程序

使用者註冊帳戶成功後，需要能夠登入帳戶，進到會員頁面。

登入功能端點：

- 網址：<http://127.0.0.1:3000/signin>
- 連線方法：POST

必須滿足的運作流程：

1. 使用者在表單中，輸入帳號 (username)、密碼 (password)，點擊【登入】按鈕。前端使用 JavaScript 檢查是否有未輸入的資訊，若有，阻止表單送出。
2. 連線到【登入功能端點】，後端接收前端傳進來的資料，並按照以下程序處理：
  - a. 檢查資料庫的 member 資料表中是否有對應的帳號、密碼。
    - i. 如果有，將會員 ID、帳號、姓名等資訊加入 Session 中紀錄，登入成功，導向到【會員頁面】，並在頁面中顯示會員姓名 (name)。
    - ii. 如果沒有，登入失敗，導向到【失敗頁面】，並顯示「[帳號或密碼輸入錯誤](#)」

#### 要求四：建立登出帳號程序

使用者登入帳戶後，可以在會員頁面點擊登出帳戶。

登出功能端點：

- 網址：<http://127.0.0.1:3000/signout>
- 連線方法：GET

必須滿足的運作流程：

1. 在【會員頁面】的後端程式，透過 Session 檢查使用者狀態，使用者狀態若未包含適當的資訊，必須直接導向回【網站首頁】，不顯示登入後的網頁內容。
2. 在【會員頁面】中加入登出連結，使用者點擊，連線到【登出功能端點】，清空 Session 中記錄的會員資訊，並導向到【網站首頁】。

### 要求五：建立簡易留言系統

在會員頁面中，加入留言功能，並且將過去網站中的留言紀錄全部顯示在頁面下方。

會員頁面：

- 網址：<http://127.0.0.1:3000/member>
- 參考設計如下：加入留言功能、簡易的顯示所有留言的作者姓名和內容。

歡迎光臨，這是會員頁

ply，歡迎登入系統  
[登出系統](#)

快來留言吧

內容

送出

丁滿：我也來試試看  
ply：這是測試留言  
ply：好哦好哦  
ply：測試測試，測試測試

留言功能端點：

- 網址：<http://127.0.0.1:3000/createMessage>
- 連線方法：POST

必須滿足的運作流程：

1. 使用者進入【會員頁面】時，後端向資料庫取得所有留言內容，將留言內容傳遞到樣板引擎中，顯示在會員頁面的下方。
2. 會員輸入留言內容 (content)，點擊【送出】按鈕。
3. 連線到【留言功能端點】，後端接收前端傳進來的資料，根據登入時在 Session 紀錄的會員 ID，將留言內容紀錄到 message 資料表。完成後導向回【會員頁面】。

### 要求六：建立刪除留言功能 (Optional)

會員可以刪除自己發表的留言。

會員頁面：

- 網址：<http://127.0.0.1:3000/member>
- 參考設計如下：自己發表的留言旁邊，顯示刪除按鈕。

歡迎光臨，這是會員頁

丁滿，歡迎登入系統  
[登出系統](#)

快來留言吧

內容

送出

丁滿：好了拉   
丁滿：哦哦哦哦，慢走不送   
精靈：我要回西方了～  
精靈：我是精靈  
丁滿：測試測試，測試測試

刪除留言功能端點：

- 網址：<http://127.0.0.1:3000/deleteMessage>
- 連線方法：POST

必須滿足的運作流程：

1. **【會員頁面】**顯示的留言中，若是目前登入會員的留言，旁邊新增**【刪除】**按鈕。
2. 會員點擊**【刪除】**按鈕，前端使用 JavaScript 內建的 `confirm` 函式重複確認是否刪除，會員若選擇取消動作，不做任何事情。
3. 會員若確定要刪除，則連線到**【刪除留言功能端點】**，後端接收前端傳入的留言 ID，根據留言的 ID 刪除對應的留言資料。完成後導向回**【會員頁面】**。