

延續上週作業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。

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：

首頁網址：<http://127.0.0.1:3000/>

參考設計如下：

The screenshot shows a web page with a dark header bar containing the text "歡迎光臨，請註冊登入系統". Below this, there are two main sections: "註冊帳號" (Registration Account) and "登入系統" (Login System). The "註冊帳號" section contains three input fields labeled "姓名" (Name), "帳號" (Account), and "密碼" (Password), followed by a "註冊" (Register) button. The "登入系統" section contains two input fields labeled "帳號" (Account) and "密碼" (Password), followed by a "登入" (Login) button.

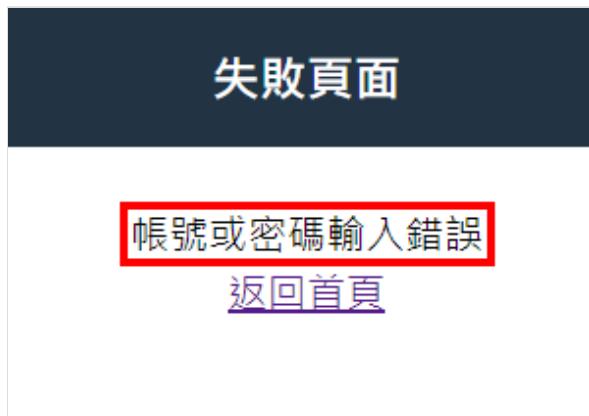
會員頁網址：<http://127.0.0.1:3000/member/>

參考設計如下：登入成功後的會員頁，要包含使用者的姓名，如下紅色部份。



失敗頁網址：<http://127.0.0.1:3000/error/?message=自訂的錯誤訊息>

參考設計如下：錯誤訊息，必須動態的從網址列的 Query String 取得，如下紅色部份。



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. 使用者輸入姓名 (name)、帳號 (username)、密碼 (password)，點擊【註冊】按鈕。
2. 連線到【註冊功能網址】，後端程式接收使用者輸入的資料，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【失敗頁網址】，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，導回【首頁網址】

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. 使用者輸入帳號、密碼，點擊【登入】按鈕。
2. 連線到【登入功能網址】，後端程式接收使用者輸入的資料，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，將使用者姓名加入 session 中紀錄，登入成功，將使用者導向【會員頁網址】，並在頁面中顯示使用者姓名。
 - ii. 如果沒有，登入失敗，將使用者導向【失敗頁網址】，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【會員頁網址】的後端程式，檢查使用者狀態，使用者狀態若為未登入，必須直接導向回【首頁】，不顯示登入後的網頁內容。
2. 在【會員頁網址】中加入登出連結，使用者點擊，連線到【登出功能網址】，在後端 session 中記錄使用者狀態為未登入，並導向回【首頁】。