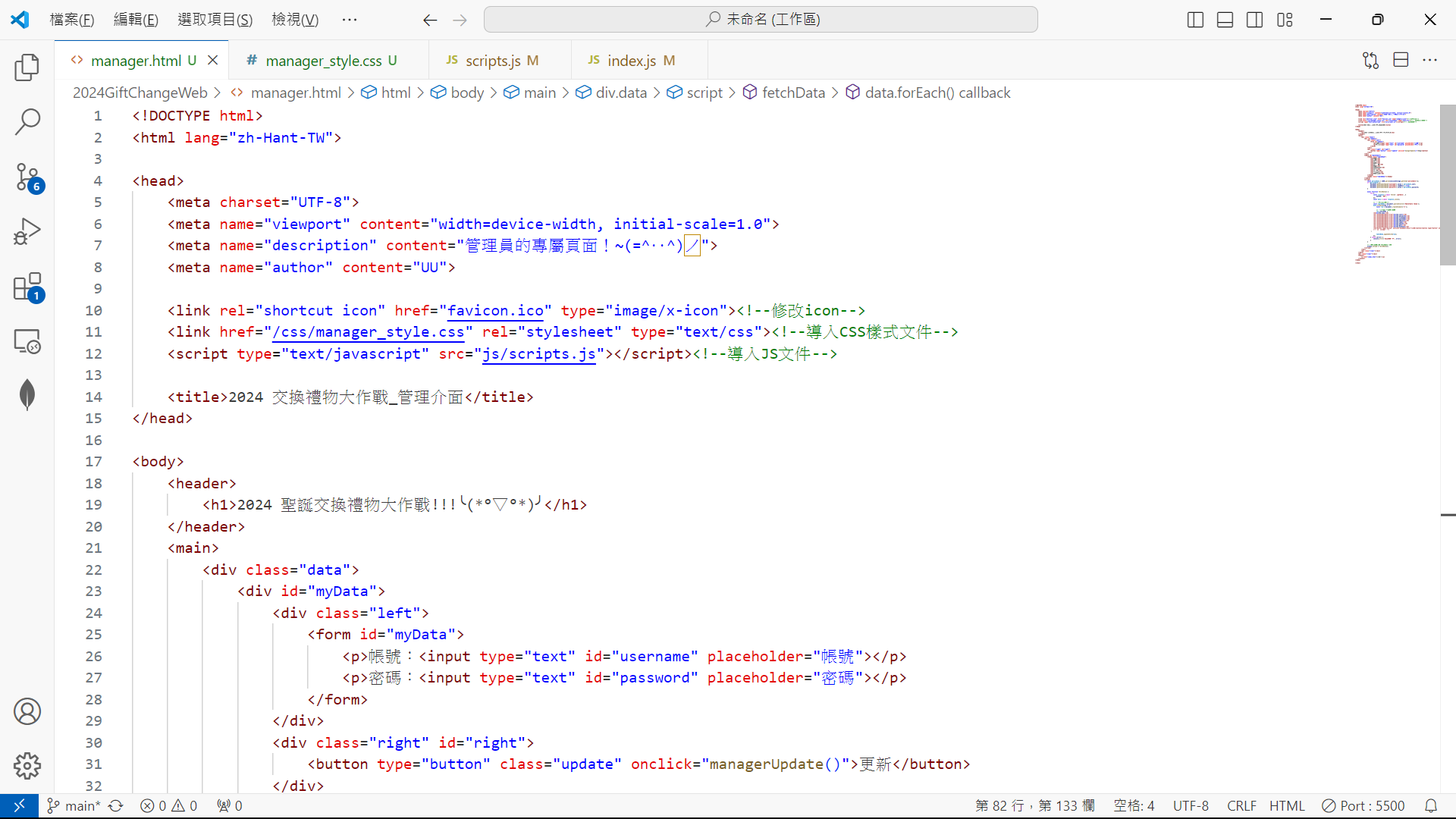
網頁製作紀錄:

09/26(四)

我宣布今天的MVP是chatGPT！(X

管理員的頁面也是我挺頭疼的，因為不知道怎麼設計比較好。好在有chatGPT，用問答的方式真的比上網搜尋還要更快且更精準的找到我要的資訊。(所以AI真的能代替程式碼工程師嗎？突然有點自我懷疑了)

1. 不管怎麼樣，管理員都是要有獨立頁面的，和一般使用者的頁面不一樣(但差不多)所以先加了一個新的頁面檔案。



2. 想了很久，還是想做到能看到全部資料，並且進行修改或刪除的功能。看到所有資料的方式，第一個想到的就是用表格(這時還沒有想怎麼寫一個可編輯的表格)所以就先把雛型建立好。

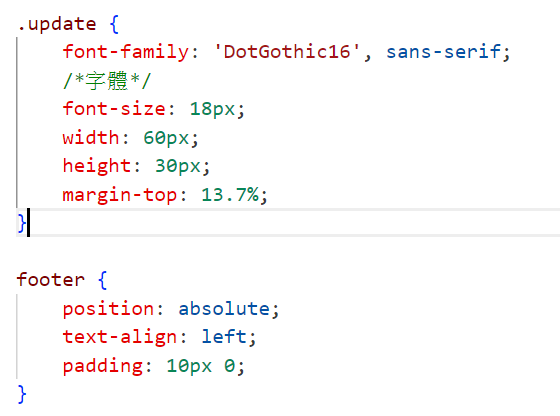


CSS也當然得獨立一個檔案，並且作細部的編輯。基本上和home的CSS差不多，只是因為有table(還有按鈕名稱有改一下)所以少數地方不太一樣。

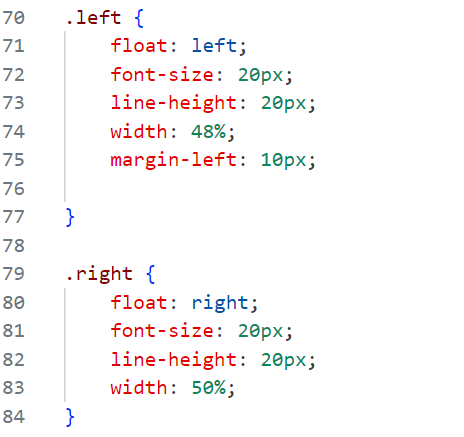
參考網址：<https://www.webdesigns.com.tw/css-table.asp>



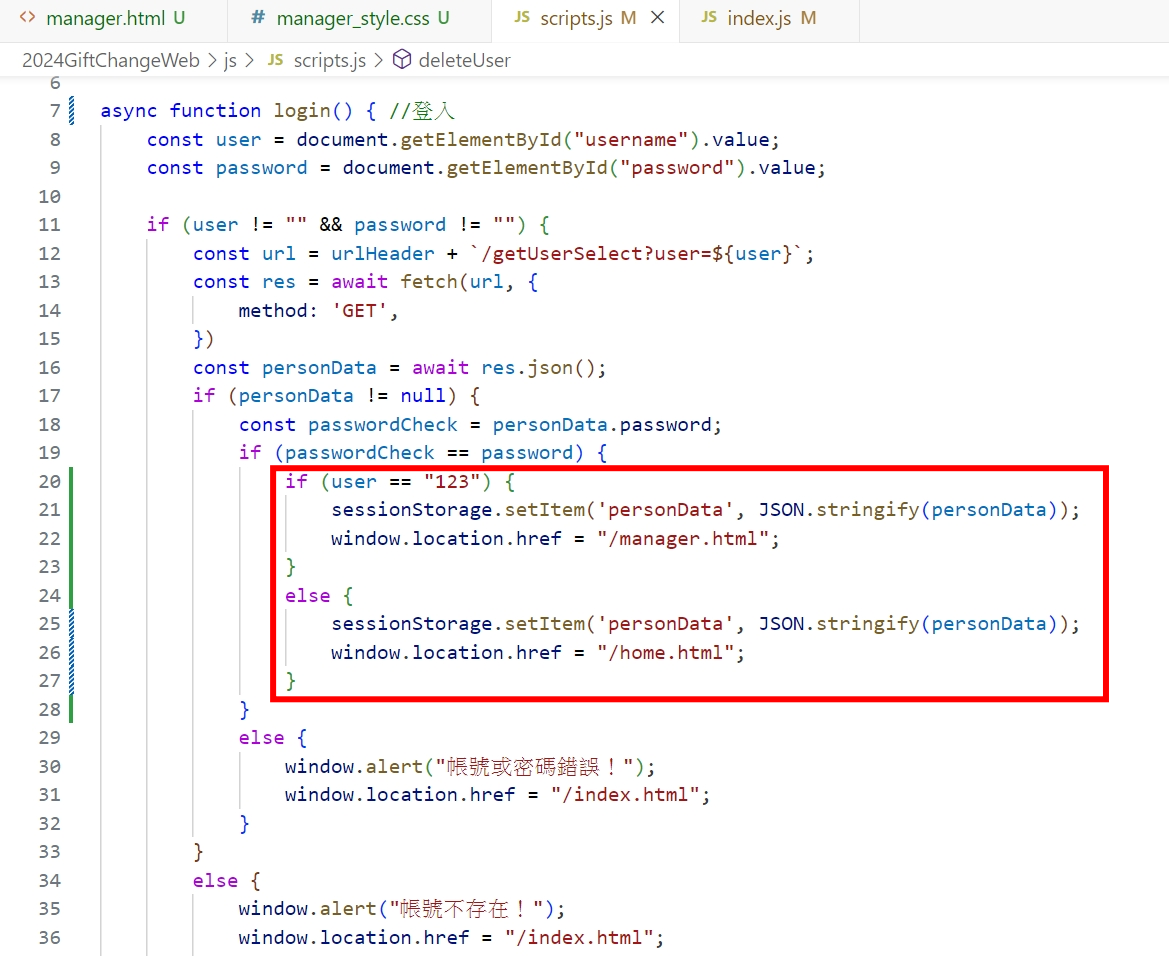
基本上有註解的地方都有chatGPT的幫助：比如讓表格的位置下去一點點、footer永遠置底。



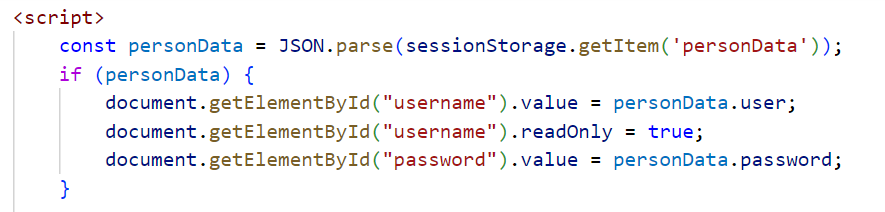
按鈕放到右側了所以也有一點點的空間排版變動，這倒是自己調的。(沒那麼了不起(X)



3. 頁面設計好之後，就要讓管理員登入之後是”管理員的專屬頁面”而不是一般的home.html了。在前端的js裡作區隔。



前端顯示管理員自己的資料，並且管理員自己可以修改自己的密碼。(帳號就別改了吧)

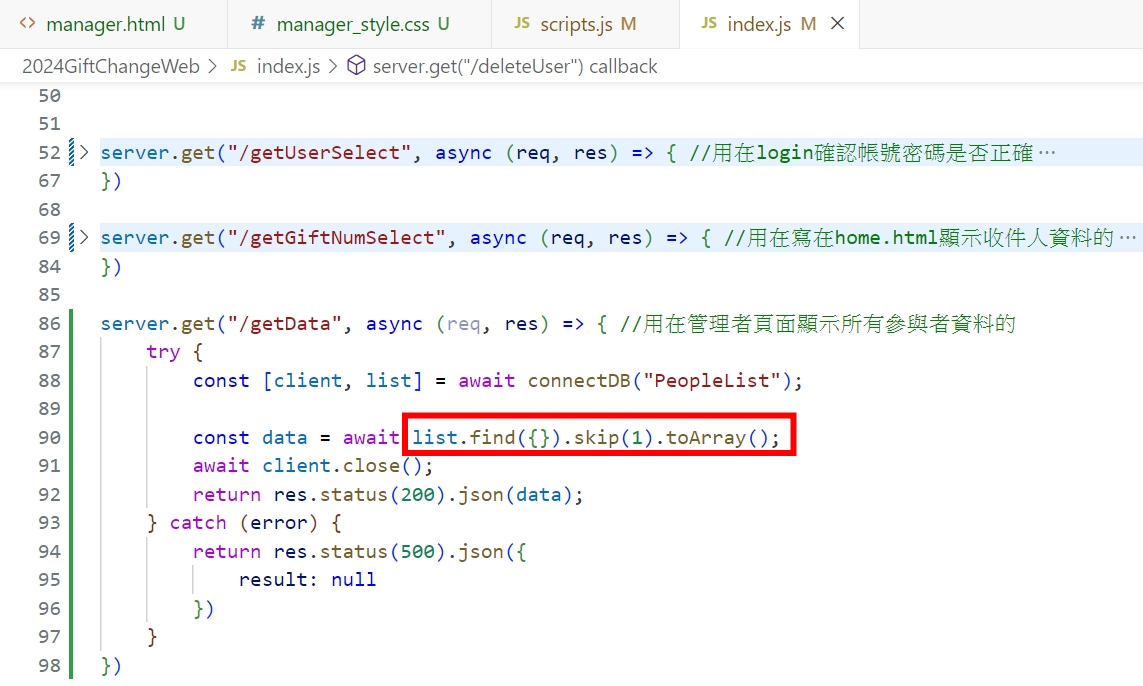






看似重複的方法好像有點多(因為我把每個方法用來幹嘛都標上了)，應該有不少是可以整併的，這個就之後有機會(應該是不會有那個機會了ㄎㄎ)的時候來重構時再說吧。

4. 再來就是大量靠chatGPT的地方了，首先第一個問題是「要如何使用node.js取得MongoDB collection中的所有資料、並在前端用js將他們以表格呈現？」根據回覆的程式碼，先在後端建立了取得所有資料的方法。(並且因為不用顯示管理員的資料，所以要跳過第一筆)



前端也和我想得差不多，是在html中用js和迴圈增加表格中的項目。(圈註的地方是我比較不擅長的→如果不靠chatGPT可能要查很久或想不到的)(抱歉我就真的不是很擅長寫HTML和js.. Orz)





5. 基礎頁面產生成功！不求調整版型到多好看了(真心覺得字體已經拯救一切了)接著是想著<tb>能否具有讓使用者可編輯的屬性。(有！萬歲！)這樣就不用像一開始那樣蠢蠢笨笨，想著把input塞進td裡了，那是真的醜。



6. 顯示的部分順利完成後就剩下最後兩個要處理的了：編輯該使用者的資料、刪除該使用者的資料。編輯的話決定鎖定編號(用來在後端查詢使用者資料)，其他都是可以透過管理員修改的，包括帳號。所以就問了怎麼用按鈕取得該欄位的所有資料。「該如何透過最後一格<td>中的按鈕與JS來保存該列資料」



取得資料之後就改寫成body的json格式，方便後續post。(chatGPT也有預料到我會這麼做！)



後端寫入更新的方法，搜尋的對象是用num進行搜尋的。(所以才會鎖定num)



7. 更新功能完成後就是刪除了，刪除要注意的是，刪除之後還要更新編號(因為編號是按著英文字母的順序來的，如果有一個缺漏抽禮物的程序就可能壞掉)，所以整體邏輯是：刪除該人員的資料→抓取所有資料(除了管理員)→根據user名稱(原本用默認的id但好像寫錯字了，導致管理員都會被改到，就換一個搜尋了)一個個更新其編號。

參考網址：<https://docs.mongoing.com/can-kao/mongo-shell-methods/collection-methods/db-collection-deleteone>



8. 最後不管更新還是刪除完都要刷新頁面。



網站全部大功告成！