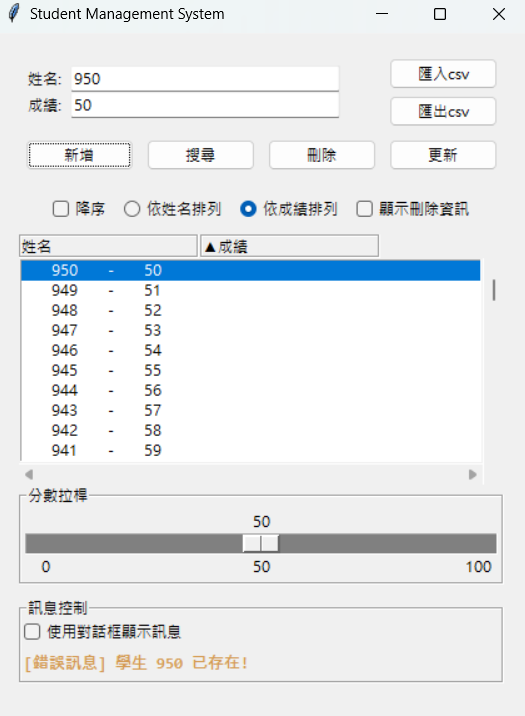
簡介

這是一個使用 python Tkinter 套件撰寫的一份程式碼，是一個學生成績管理系統，支援csv檔案格式用於儲存與導入資料。



圖二、學生成績管理系統 (Listbox)



圖一、學生成績管理系統 (Treeview)

功能介紹

1. **增加學生:**「新增」按鈕，將當前輸入框 (姓名和成績) 當中的資料顯示道下方的展示區當中，假如輸入的資料有誤 (未輸入資料、分數範圍不對、僅輸入名字等)，會跳出messagebox，顯示錯誤訊息提醒使用者。
2. **搜索學生:**「搜尋」按鈕，搜索當前姓名輸入框內容的學生資料。
3. **刪除學生:**「刪除按鈕」，刪除當前姓名輸入框內容的學生資料，刪除成功的學生會被標記為「已刪除」，此時勾選「顯示刪除資訊」能夠顯示被刪除的學生資料。
4. **更新學生:**「更新按鈕」，更新當前姓名輸入框內容的學生資料。

除了新增成功之外，所有行動成功與否，都會透過messagebox告知使用者，學生不存在或者沒有輸入內容，都會跳出對應的訊息。

1. **滾動條(Scrollbar):** 資料顯示區域在資料過多的時候，會自動產生兩個方向的滾動條 (x方向和y方向)。
2. **分數篩選(Scale):** 視窗最下方有一個分數篩選的滑桿，當學生分數低於滑桿的數值，就不會出現在資料顯示區域。
3. **排序依據(RadioButton):** 在學生增刪改查的功能下方，有幾個按鈕能選擇當前的排序規則，能依據姓名，或者分數排序，以及是要升序還是降序。
4. **檔案匯入匯出:** 學生資料在編輯完成之後，可以透過「匯出csv」去將資料用csv格式匯出，同樣可以透過「匯入csv」將當前的資料覆蓋掉。
5. **使用者體驗優化:** 加入了一些提高使用效率的功能，像是雙擊某個學生，就能將它的資料填到輸入框當中，方便後續的操作，點擊標題欄位，能更動排序規則，右鍵則是除此之外，一併更動順序或倒序。雙擊某個學生，就能將它的資料填到輸入框當中，方便後續的操作，
6. **訊息控制:** 「訊息控制區塊」能設定要不要使用 messagebox 顯示資訊，假如打開，就使用 messagebox，否則將訊息顯示在下方的區塊。
7. **摸魚:** 「更新」按鈕被滑鼠中鍵按下之後，會開啟子程式，用 tkinter 復刻的簡單踩地雷遊戲。

結論

這份作業把上課的內容混在一起，用一個小專案的方式確認自己有沒有學會 tkinter，在寫的過程當中不太遇到對語法不熟悉的問題，畢竟上課有一半的時間都在寫，真的不會就問問AI，它總能蹦出一些我沒看過的寫法，所以在寫的過程中也漸漸學到了許多 python 才有的特別用法。

最後我決定使用類別去建立整個程式，這並不是一時興起，而是在最初上課單用函數的方法總能遇到一些問題，而這些問題在換成類別之後全都消失了，因此在練習的中後期，每一份作業都使用類別，漸漸地也寫出了一些固有的規則，主函數都是簡單的兩行，產生一個該類別的物件，然後透過.start()啟動它，類別中則都是 create\_widget()、create\_variable()，按鈕事件都用on\_為開頭，很快地做出了增刪改查的功能。

此時的介面使用listbox顯示，但這實在太醜了，所以我換成了tkinter.ttk當中的 treeview，介面變得好看了些，不過這時就出現了另一個問題，x軸的 scrollbar 沒辦法用了，因為treeview會確保每一欄都被顯示出來，當加入過長的名字之後無法調整，因為極限取決於其他欄的最小寬度，最後又換回 listbox 的版本 (說是這樣說，但最後兩個都寫了)

在上課時老師有說過過去的學生做出了檔案的匯入和導出，因為我沒什麼創意，就將這個想法偷了過來，首先要將資料不只顯示在資料展示區當中，所以更動到資料內容的部分都要重寫，不過因為前面打了良好的基礎，耦合度低，所以很快就建立出另一個類別StudentData，將所有動到檔案的功能通通封裝進去，再加入兩個按鈕，支援檔案的輸出輸入，功能就完成了，不過沒有加入自動導入資料的功能，因為這只節省了多按兩下按鈕的時間。

那麼那些動作是最花時間的? 很明顯就是在測試時經歷無數次的按鈕，每次要測試某個特定學生時，都需要輸入姓名，輸入分數，因為有輸入限制的關係，還不能亂打，因此加入了雙擊將學生資料顯示在輸入框的功能。除此之外，因為排序加入了順序和倒序兩種型態，點RadioButton有些浪費時間，因此更改為滑鼠左右鍵就能直接修改，加快了使用的效率。

第二花時間的，就是把冒出來的 messagebox 關掉的時間，所以我將 messagebox 封裝了一層，用開關讓使用者能設定要不要打開 messagebox 的通知功能，關掉的話，則會以另一種方式顯示在畫面中。

最後，是我私心想要加入的東西，也就是在一個月前寫好的踩地雷閹割版，觸發條件是用滑鼠中鍵點擊更新兩次觸發，這對學生成績管理系統完全是沒必要的東西，但我還是加進去了，因為這很有趣。

在寫的過程當中，每寫一個段落就又會有一些問題冒出來，像是「如果資料量太大，每次刷新都會很花時間，有沒有更好的刷新方法?」，「csv檔案在匯入的時候，是要加入到後面還是直接覆蓋? 還是再加一個按鈕?」「保存的csv 要不要將是否刪除一併加入，還是只存沒被刪掉的?」「要不要加入自動存檔功能?」，但都加進去就變得太多了，寫不完，最後的版本有 listbox, treeview 兩個版本，因為不清楚上課沒教到的 treeview 能不能算用「tkinter」完成，實在太方便，太好用了，方便的心理犯怵，像開了外掛一樣。