

Homework 2 個人記帳系統

Deadline: 2022-10-24 23:59:59

前言

近期物價上漲，小明發現在月底之前他的錢就不知不覺沒了，由於他沒有對每筆支出和收入做紀錄，以至於小明不確定自己到底在哪邊花特別多的錢，也不確定自己可以從哪邊開始著手省錢，他希望同學可以運用 Java 程式語言幫助自己建立一個個人記帳系統，讓他可以記錄與查詢自己這一個月以來的收支狀況。

作業需求

■ Functional Requirements

- 使用者可以透過指令來進行帳目的新增、查詢、修改、刪除。
- 此記帳系統可以記錄至多 100 筆帳目資料。
- 新增的每一筆帳目包含日期（1~31 日）、類別、金額。
- 帳目類別包含娛樂、飲食、交通，並以代碼表示，如下表：

類別	代碼
娛樂	a
飲食	b
交通	c

- 基本指令集：

操作	指令格式	說明
新增	add [日期] [類別] [金額] 範例：add 4 a 150	新增一筆帳目。 新增一筆 4 日娛樂類的消費
查詢	show all show [日期] 範例：show 20 show [類別] 範例：show c	搜尋所有帳目。 搜尋於特定日期的所有帳目。 搜尋 20 日的所有帳目。 搜尋特定類別的所有帳目。 搜尋交通類的所有帳目。
修改	update [編號] [日期] [類別] [金額] 範例：5 15 b 200	進入修改模式。 修改該編號的帳目。 將編號五的帳目修改為 15 日飲食類別的 200 元消費。
刪除	delete	進入刪除模式。
幫助	help	列出基本指令集，並說明各指令功能。
離開	exit	結束系統。

[註]：

1. 所有指令均是小寫或阿拉伯數字，指令和指令中間以半形符號的空白鍵區隔。
2. 在批改作業時，助教會測試程式各個指令是否能正常執行來給予分數，但指令間是具

有相依性的，例如需要**新增**功能，才能有 data 對其餘功能進行測試，**查詢**功能需正常運作，才能測試**修改與刪除**功能。因此同學在撰寫作業時，對於這兩個功能要特別留意是否能正常執行，否則會影響到其他功能的給分。

3. 指令的輸入可以運用 String 的 method: split() 實作，請參考 hw2 附加檔案。
4. 當使用者輸入**錯誤指令(包含拼字、格式、數字範圍錯誤等)**，跳出**錯誤提示**並要求使用者**重新輸入**。

■ Non-functional Requirements

- 提供完善、人性化、易閱讀的使用者介面。
- 當使用者輸入指令後，需有相對應的回應，提供良好的使用者體驗。
- 由於作業評分的需要，**不可**擅自增加額外的停頓點或延遲效果。

■ Execution Flow (Use Case)

1. 執程式後，顯示歡迎訊息，同時提示使用者可以輸入的指令集。
2. 讓使用者輸入欲使用的指令。
3. 在 2. 中使用者若輸入 add [日期][類別][金額]，系統須顯示使用者是否 add 成功，若成功則須顯示 add 後的結果，失敗則須顯示失敗原因，無論失敗或成功皆回到 2. 讓使用者輸入指令。
4. 在 2. 中使用者若輸入 update，先將目前計帳系統中所有項目列出並編號(編號從 1 開始，順序:依日期順序由小到大排序，同日期中的資料再依加入記帳本的順序排序)，讓使用者能夠利用編號選擇 update 的項目，接著讓使用者輸入修改的指令，系統須顯示使用者是否 update 成功，若成功則須顯示 update 後的結果，失敗則須顯示失敗原因，無論失敗或成功皆回到 2. 讓使用者輸入指令。
5. 在 2. 中使用者若輸入 delete，先將目前計帳系統中所有項目列出並編號(編號從 1 開始，順序:依日期順序由小到大排序，同日期中的資料再依加入記帳本的順序排序)，讓使用者利用編號選擇 delete 的項目，接著讓使用者輸入欲刪除的編號，若成功則須顯示 delete 後的結果，失敗則須顯示失敗原因，無論失敗或成功皆回到 2. 讓使用者輸入指令。
6. 在 2. 中使用者若輸入 show [日期]，依加入記帳本的順序排序顯示該日期所有項目，若查無任何資料則顯示查無資料，顯示完畢回到 2. 讓使用者輸入指令。
7. 在 2. 中使用者若輸入 show [類別]，依日期順序由小到大排序，同日期中的資料再依加入記帳本的順序排序顯示該類別所有項目，若查無任何資料則顯示查無資料，顯示完畢回到 2. 讓使用者輸入指令。
8. 在 2. 中使用者若輸入 show all，依日期順序由小到大排序，同日期中的資料再依加入記帳本的順序排序顯示所有已記帳項目，若查無任何資料則顯示查無資料，顯示完畢回到 2. 讓使用者輸入指令。
9. 在 2. 中使用者若輸入 help，顯示出所有指令並說明各指令功能，顯示完畢回到 2. 讓使用者輸入指令。
10. 在 2. 中使用者若輸入 exit，退出系統。
11. 若指令錯誤，顯示錯誤結果，並且回到 2. 讓使用者繼續輸入指令。

[註]: 成功 add、update、delete 後的結果均依日期順序由小到大排序，同日期中的資料再依加入記帳本的順序排序顯示所有已記帳項目。

額外加分項目

- 在不破壞執行流程的限制下，可為此系統加入新功能或新特色以獲得額外加分，**但需在 readme.txt 中明確說明**，包含如何觸發或使用等等，**且不可增加額外中斷點**。
 - 舉例：如果程式要求輸入 1 或 2，來決定接下來的執行步驟，則可以多增加一個選項(例如 3)做為加分項目，但是不要再輸入 1 或 2 後，額外增加更多輸入。
- User Interface 排列整齊且精美者，將能獲得額外加分。
- 加分的前提為**基本功能都正確實作**。

作業繳交方式與內容

請參考計概網站上的「[作業繳交說明與規範](#)」將作業相關檔案上傳至「hw2」當中。作業相關檔案分為以下四部分，括號中的數字為該部分的配分比：

- A. **程式原始碼檔案 (60%)**：以 JAVA 根據作業要求撰寫而成的檔案，主程式檔名為「**hw2.java**」，主程式需撰寫註解幫助別人看懂你的程式碼，若同學的程式執行時需要用到其他檔案，如「ConsoleIn.java」，也需作為程式檔案的一部分一起繳交上來。
- B. **文件檔案(22%)**：檔名為「document.doc」、「document.docx」或「document.pdf」，文件需包含以下內容：
- (1) 需求描述：描述使用者在使用此程式時可能會有何需求？你設計程式時如何考慮這些需求？程式中有哪些地方特別吸引使用者？
 - (2) 程式流程：說明程式進行流程，建議使用利用流程圖或是其他有助說明的圖示來幫助說明程式整體上如何運作。
 - (3) 功能/邏輯說明：說明自己的程式裡，自己針對什麼樣的「功能」特地設計了什麼樣特別的「邏輯」來達成？建議用畫圖的方式來說明。
 - (4) 使用說明：以教導使用者的角度，說明程式要如何執行使用或是需特別注意的點...等等（不可只寫「看了就會使用」之類的話）。
 - (5) 其他：任何有助於別人了解或使用你的程式的說明。切勿流於閒聊。
- C. **readme.txt (3%)**：其中必須描述以下內容
- (1) 作者：系級、修課班別、學號、姓名。
 - (2) 程式檔案名稱：用來完成作業需求的程式檔案名稱。
 - (3) 使用說明：以教導使用者的角度，說明程式要如何執行使用，操作注意事項、附加功能以及其觸發方式...等等。
- D. **TimeLog.doc**：記錄此次作業撰寫時間（「確實」填寫該表的同學可獲加 5 分），請參考計概網站上的「[TimeLog 範本](#)」。

特別注意事項

1. 繳交作業請使用計概網站上傳，若有相關問題請參考計概網站上的[作業繳交說明與規範](#)。
2. 請務必確保檔案擺放及命名符合規定，若作業檔案放錯資料夾、檔名大小寫或拼寫有誤，將可能造成作業無法正確批改。因此，**出現此情形將酌情扣分**。
3. **作業請勿抄襲，所有作業皆會經過程式比對判斷是否抄襲，若發現一律以零分計算。**