小型帳務軟體使用說明書

作者：Rick Lin

**目錄**

[**目錄** 2](#_Toc103543775)

[Chap1 系統簡介 3](#_Toc103543776)

[Chap2 初次使用環境設定 4](#_Toc103543777)

[**2.1** **修改使用者資訊** 4](#_Toc103543778)

[**2.2** **修改帳冊資訊** 4](#_Toc103543779)

[**2.3** **設置期初庫存與初始日期** 5](#_Toc103543780)

[Chap3更新版本流程 6](#_Toc103543781)

[**3.1** **解壓縮新版本** 6](#_Toc103543783)

[**3.2** **將舊資料庫覆蓋至新路徑下** 6](#_Toc103543784)

[**3.3** **更新資料庫物件與資料** 6](#_Toc103543785)

[Chap4 功能說明 7](#_Toc103543786)

[**4.1** **FUN1.1 一般交易** 7](#_Toc103543787)

[**4.2** **FUN1.2 銀行存提** 9](#_Toc103543788)

[**4.3** **FUN1.3 結帳** 10](#_Toc103543789)

[**4.4** **FUN2.1 交易明細表** 11](#_Toc103543790)

[**4.5** **FUN2.2 庫存明細表** 12](#_Toc103543791)

[**4.6** **FUN2.3 經常費收支表** 13](#_Toc103543792)

[**4.7** **FUN2.4 會計師財務報表** 14](#_Toc103543793)

[**4.8** **FUN3.1 帳冊基本資料維護** 15](#_Toc103543794)

[**4.9** **FUN3.2 管理員命令工具** 16](#_Toc103543795)

Chap1 系統簡介

**功能：**提供持有「銀行存款」與「現金」的使用者作帳，產出報表。

**架構：**採用WPF與SQLite搭配Dapper實作

**結帳流程：**輸入一般交易 >> 輸入銀行存提 >> 結帳 >> 列印報表

1. 交易：主要分為「一般交易」與「銀行存提」，前者為一般性的收入支出，只記錄單一方向的交易，後者為不同帳本間的調撥，記錄雙向交易，例如銀行提款，為銀存帳本支出，現金帳本收入。
2. 結帳方式：為「月結」，每個月結帳一次。
3. 報表：分為「系統報表」及「呈報報表」。
4. 管理員命令工具：提供管理員下SQLite指令處理問題，Select或異動皆可。

Chap2 初次使用環境設定

**若您為第一次使用，請參照以下流程完成初始設定**

* 1. **修改使用者資訊**

於主畫面編輯「使用者、使用單位」，按下【更新使用資訊】



* 1. **修改帳冊資訊**

1. 於「主選單>>設定>>帳冊基本資料維護」，編輯帳冊資訊
2. 銀存帳冊：輸入「銀行相關欄位」資料(銀行代碼、名稱、帳號、戶名)
3. 帳冊名稱：可自行修改，方便識別

PS.系統預設為「總帳冊、銀存帳冊、現金帳冊」各一，若有需要多個銀存或現金帳本，可自行新增



* 1. **設置期初庫存與初始日期**

先假定好您要開始的日期，建議是年初，因為有年度報表會用到，下面提供一個範例

Ex.

假定系統由2022/1/1開始作帳，所以需要2021/12/31的庫存如下

|  |  |  |  |
| --- | --- | --- | --- |
| 帳冊 | 帳冊代碼(來自帳冊資料) | 庫存日 | 庫存數 |
| 總帳冊 | Total | 2021-12-31 | 10000 |
| 銀行存款 | Bank | 2021-12-31 | 6000 |
| 現金 | Cash | 2021-12-31 | 4000 |

1. 開啟「設定>>管理員命令工具」
2. 依照設定的期初庫存，修改以下紅字部分，全選並貼上，按下【執行】

注意：「'」或「,」都不可忽略，日期請用「yyyy-MM-dd」的格式(ex.2021-01-31)

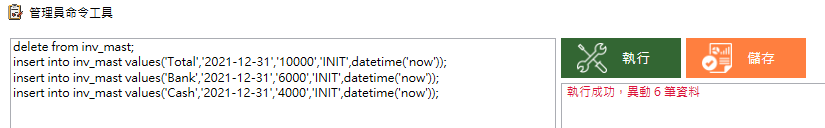
如果您前面除了基本三個帳冊有新增帳冊，語法需要增加一行賦予它起始庫存

delete from inv\_mast;

insert into inv\_mast values('Total','2021-12-31','10000','INIT',datetime('now','localtime'));

insert into inv\_mast values('Bank','2021-12-31','6000','INIT',datetime('now','localtime'));

insert into inv\_mast values('Cash','2021-12-31','4000','INIT',datetime('now','localtime'));



1. 依照設定的初始作帳日，修改以下紅字部分，全選並貼上，按下【執行】

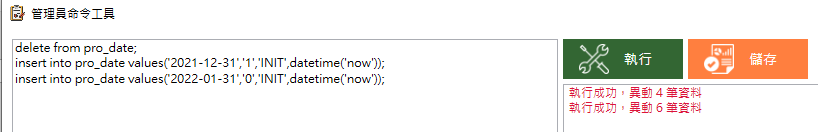
以本範例而言，需要新增一筆2021-12-31已結帳的資料，以及一筆2022-01-31未結帳的資料，所以會有兩筆(一筆為1代表已結帳，一筆為0代表未結帳)

注意：「'」或「,」都不可忽略，日期請用「yyyy-MM-dd」的格式(ex.2021-01-31)

delete from pro\_date;

insert into pro\_date values('2021-12-31','1','INIT',datetime('now','localtime'));

insert into pro\_date values('2022-01-31','0','INIT',datetime('now','localtime'));



**至此設定完成，可以開始作帳!**

Chap3更新版本流程



**若您已有系統，只更新版本，請依照以下流程更新，避免資料遺失**

由於本系統為離線操作，帳務資料儲存於您的電腦，檔案為系統資料夾下的「Accounting\_App.db」，有需要可將其備份，以下說明更新方式。

* 1. **解壓縮新版本**

解壓縮新版本至一個新的路徑，切勿直接覆蓋原本路徑，否則「Accounting\_App.db」被覆蓋後資料將遺失。

* 1. **將舊資料庫覆蓋至新路徑下**

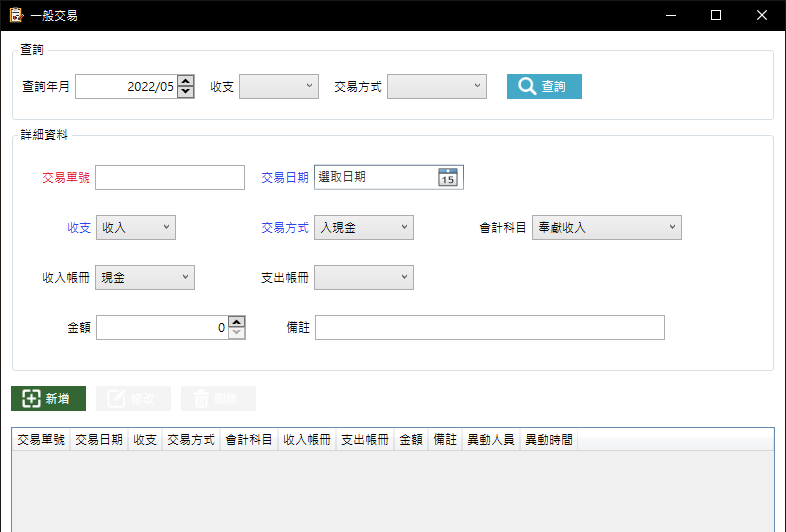
將您舊路徑下的「Accounting\_App.db」複製，貼到新路徑下，覆蓋檔案。

* 1. **更新資料庫物件與資料**

若本次更新有特別說明需要執行的項目，再依序執行，沒有可忽略。

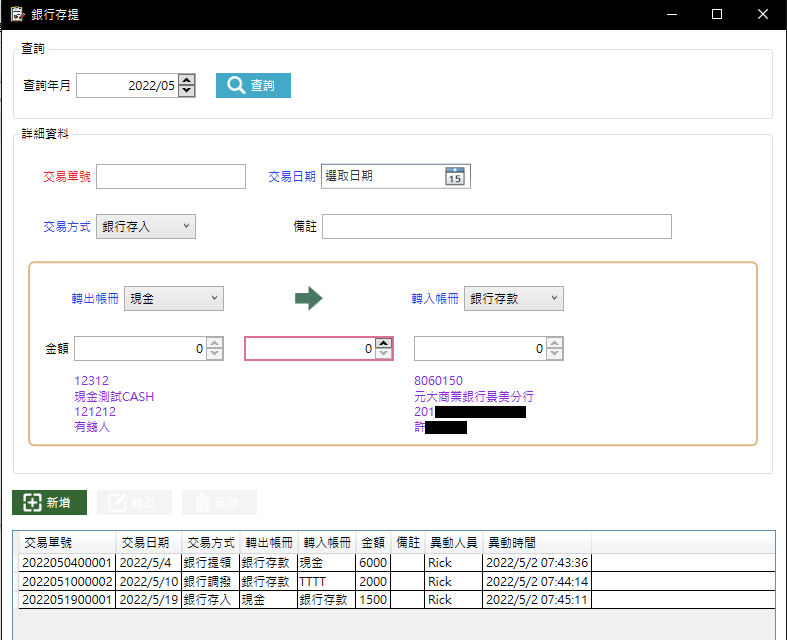
Chap4 功能說明

* 1. **FUN1.1 一般交易**



|  |
| --- |
| 基礎流程 |
| 查詢：選擇查詢資料，按下【查詢】  新增、修改：按下【新增】或【修改】，於[詳細資料]編輯後，按下【確定】  刪除：選擇資料後，按下【刪除】，按下【確定】 |
| 收支欄位連動 |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | **收支** | **交易方式** | **會計科目** | **收入帳冊** | **支出帳冊** | | 收入 | 入現金 | 奉獻收入  存款息 | 帳冊類型為「現金帳冊」的帳冊 | 無 | | 入銀存 | 帳冊類型為「銀存帳冊」的帳冊 | | 支出 | 現金 | 薪資  水電  郵電費  文具印刷  保險費  愛餐費  對外奉獻  交通費  什項購置  圖書雜誌  外請講員  修繕維護  設備購置  慶弔關懷費  教育訓練  什支  稅捐  教師鐘點費  建堂 | 無 | 帳冊類型為「現金帳冊」的帳冊 | | 銀行轉帳 | 帳冊類型為「銀存帳冊」的帳冊 | | 其他 | 「現金帳冊」和「銀存帳冊」 | |
| 新增、修改、刪除前的檢核 |
| 1. 若當月已經結帳，顯示「X月已經結帳，不可修改」 |
| 新增、修改、刪除後的檢核 |
| 1. [交易日期]必填 2. [金額]不可為0 3. 若當月已經結帳，顯示「X月已經結帳，不可修改」 |

* 1. **FUN1.2 銀行存提**



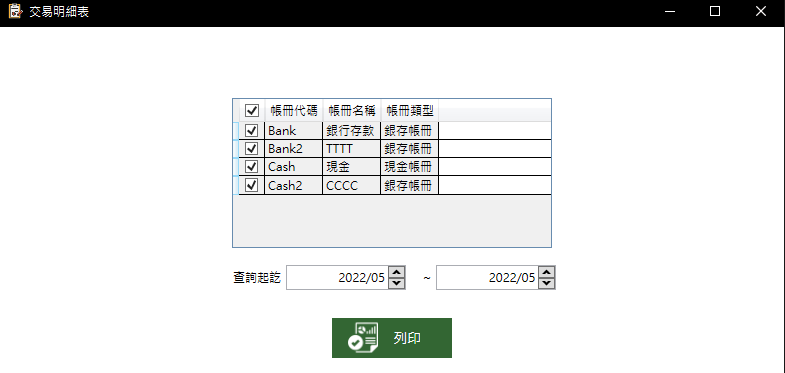
|  |
| --- |
| 基礎流程 |
| 查詢：選擇查詢資料，按下【查詢】  新增、修改：按下【新增】或【修改】，於[詳細資料]編輯後，按下【確定】  刪除：選擇資料後，按下【刪除】，按下【確定】 |
| 交易方式欄位連動 |
| |  |  |  | | --- | --- | --- | | **交易方式** | **轉出帳冊** | **轉入帳冊** | | 銀行存入 | 帳冊類型為「現金帳冊」的帳冊 | 帳冊類型為「銀存帳冊」的帳冊 | | 銀行提領 | 帳冊類型為「銀存帳冊」的帳冊 | 帳冊類型為「現金帳冊」的帳冊 | | 銀行調撥 | 帳冊類型為「銀存帳冊」的帳冊 | 帳冊類型為「銀存帳冊」的帳冊 | | 現金調撥 | 帳冊類型為「現金帳冊」的帳冊 | 帳冊類型為「現金帳冊」的帳冊 | |
| 帳冊資訊與金額連動 |
| 選擇[轉入帳冊]或[轉出帳冊]系統自動做以下連動   1. 帶出「帳冊基本資料」的銀行資訊於[金額]下方。(代號、名稱、帳號、戶名) 2. 計算當前餘額於[金額]欄位   新增時公式：  修改時公式：   1. 修改中間交易[金額]時，[轉入帳冊]與[轉出帳冊]自動加減金額 2. 自動檢核並顯示紅字訊息於畫面警示：   轉出帳冊與轉入帳冊不可相同 (因為同一帳冊調撥沒有意義)  出入帳冊庫存不可小於0 (不可以執行後造成庫存小於0) |
| 新增、修改、刪除前的檢核 |
| 1. 若當月已經結帳，顯示「X月已經結帳，不可修改」 |
| 新增、修改、刪除後的檢核 |
| 1. [交易日期]必填 2. [金額]不可為0 3. 轉出帳冊與轉入帳冊不可相同 4. 出入帳冊庫存不可小於0 5. 若當月已經結帳，顯示「X月已經結帳，不可修改」 |

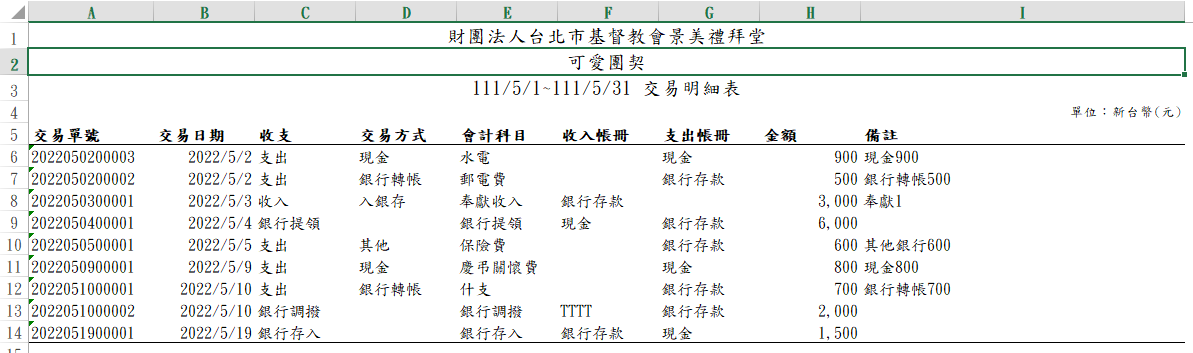
* 1. **FUN1.3 結帳**



|  |
| --- |
| 基礎流程 |
| 結帳：選擇[結帳]，按下【結帳】  反結帳：選擇[反結帳]，按下【反結帳】 |
| 畫面連動 |
| 1. 結帳：   選擇[結帳]，按鈕顯示[結帳]，畫面帶出下一結帳日，[結帳狀態]顯示「未結帳」   1. 反結帳：   選擇[反結帳]，按鈕顯示[反結帳]，畫面帶出上一結帳日，[結帳狀態]顯示「已結帳」 |
| 結帳檢核 |
| 1. 系統為「月結」，從月初至月底逐日計算帳冊庫存，若某日發生庫存小於0，顯示「XX帳冊於yyyy-MM-dd結帳時發生庫存小於0 ($金額)，請檢查交易」 |
| 結帳邏輯 |
| 結帳：   1. 先刪除大於等於結帳月的庫存 2. 每筆交易會記錄由X帳冊出Y帳冊入，用上個月底庫存對各帳冊逐日計算庫存，若當中發生庫存小於0會發警示並跳出 3. 計算完各帳冊，會加總金額，新增一筆總帳冊庫存 4. 庫存為0的帳冊會刪除 5. 結帳結束，新增一筆當月已結帳，以及一筆下個月未結帳，兩筆結帳紀錄   反結帳：   1. 還原上個月結帳紀錄 |

* 1. **FUN2.1 交易明細表**





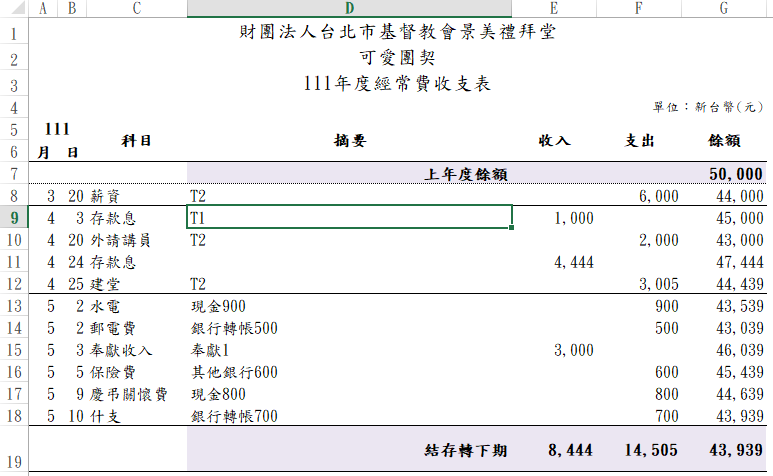
* 1. 內容：系統儲存之交易明細，參考「FUN1.1 一般交易」
  2. 帳冊：不含總帳冊的所有帳冊，多選請按住Ctrl
  3. **FUN2.2 庫存明細表**





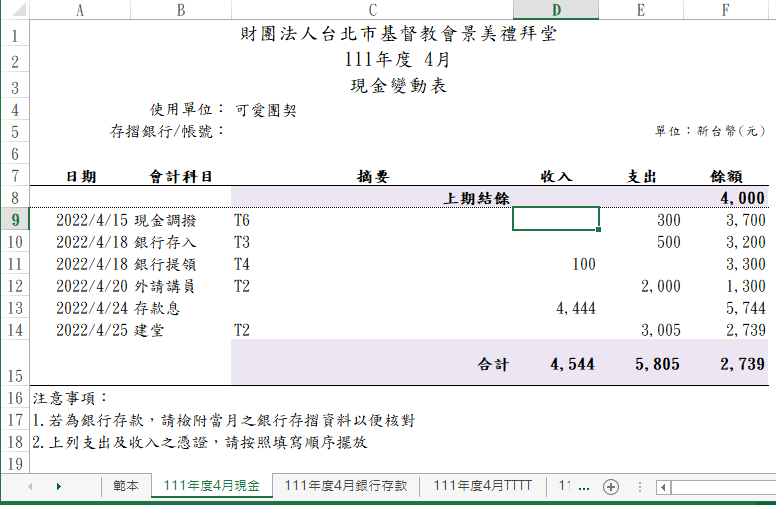
* 1. 內容：系統儲存之庫存明細，總帳冊用特殊顏色表示
  2. 帳冊：所有帳冊，多選請按住Ctrl
  3. **FUN2.3 經常費收支表**





* 1. 內容：當年度總帳收支明細，不含「FUN1.2 銀行存提」交易，僅顯示單純收支
  2. 上年度餘額：去年底庫存
  3. **FUN2.4 會計師財務報表**





1. 內容：提供會計師所需報表，交易為所有交易
2. 依照「年、月、帳冊(不含總帳)」拆分sheet
3. 上期結餘：上個月庫存
4. 存摺銀行/帳號：現金顯示空白，銀存帶出銀行名稱與帳號
   1. **FUN3.1 帳冊基本資料維護**



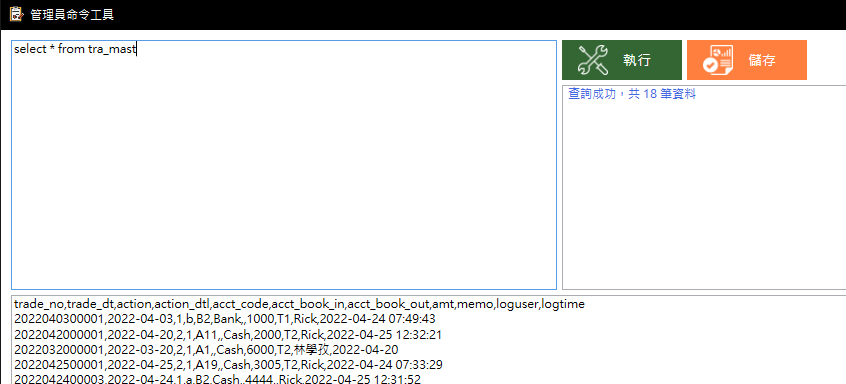
|  |
| --- |
| 基礎流程 |
| 查詢：選擇查詢資料，按下【查詢】  新增、修改：按下【新增】或【修改】，於[詳細資料]編輯後，按下【確定】  刪除：選擇資料後，按下【刪除】，按下【確定】 |
| 特殊說明 |
| 1. 本系統只允許一個「總帳冊」，為內建不可修改 2. 可新增修改現金或銀存分帳冊 |
| 刪除前的檢核 |
| 1. 若帳冊已經結帳，顯示「XX帳冊已經結帳，不可刪除」   PS.系統允許修改銀行相關資訊、帳冊名稱，但帳冊代號、帳冊類型不可修改 |
| 新增、修改、刪除後的檢核 |
| 1. [帳冊代號]是否存在 2. [帳冊代號]不可空白 |

* 1. **FUN3.2 交易備註預設值維護**



|  |
| --- |
| 基礎流程 |
| 查詢：選擇查詢資料，按下【查詢】  新增、修改：按下【新增】或【修改】，於[詳細資料]編輯後，按下【確定】  刪除：選擇資料後，按下【刪除】，按下【確定】 |
| 特殊說明 |
| 1. 預設值使用於「一般交易、銀行存提」，欄位連動參考以上功能 2. 預設值查找邏輯：輸入交易時，找最多欄位吻合的選項，若同欄位數吻合，找最下層級吻合的選項   Ex.輸入「收入、入現金、全選、現金、全選」，查找最多欄位數吻合的選項    Ex.輸入「全選、現金、薪資、全選、現金」，同欄位數找最下層級吻合的選項 |
| 新增前的檢核 |
| 1. 「收支、交易方式、會計科目、收入帳冊、支出帳冊」不可皆為空 2. 「收支、交易方式、會計科目、收入帳冊、支出帳冊」為Key值只能有一筆 |

* 1. **FUN3.3 管理員命令工具**





* 1. 提供管理員下SQLite的Select與異動語法
  2. 若為Select開頭，於下方顯示資料並顯示筆數於右方，按【儲存】可將下方資料匯出CSV檔
  3. 若不為Select開頭，執行語法，並顯示筆數於右方