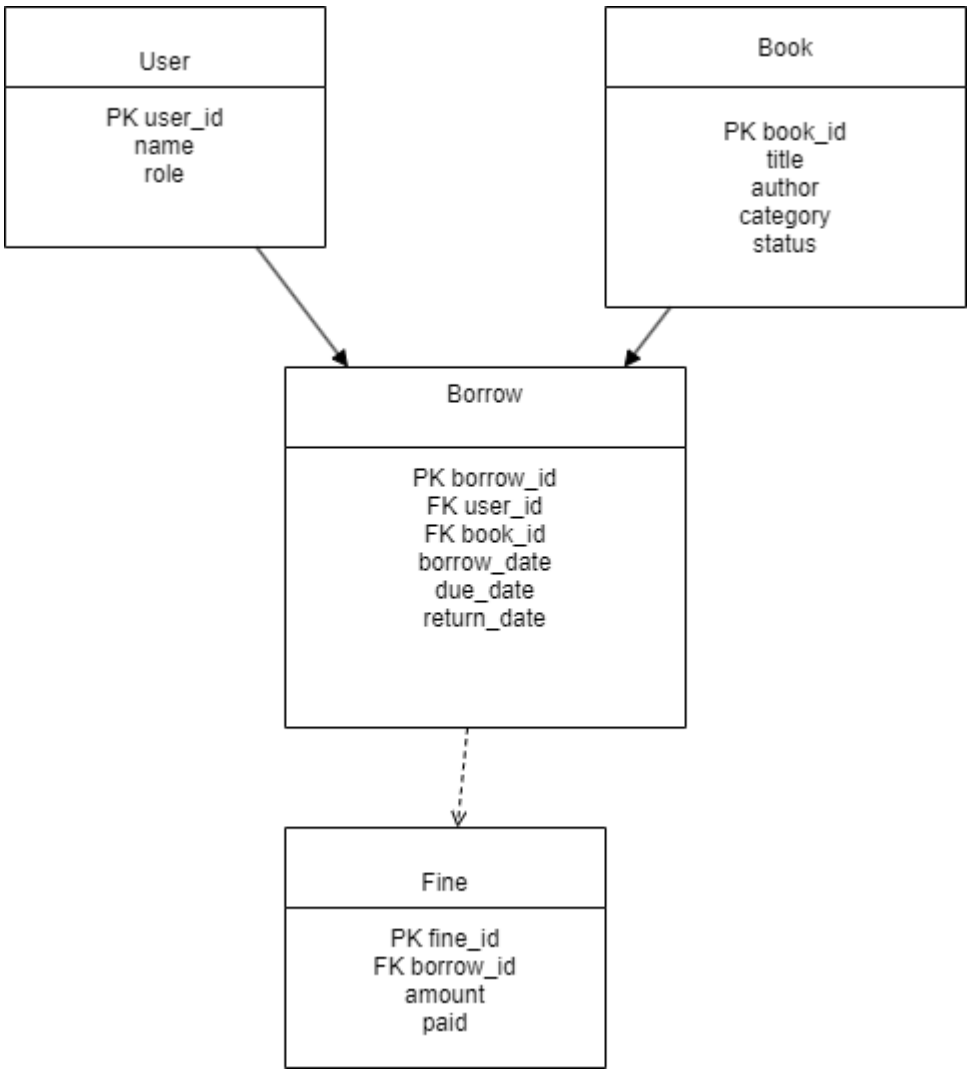
圖書管理系統

財金 3B 411530503 吳沛宸

一、 系統功能構想

借書、還書、逾期產生罰金、圖書查詢、借閱統計查詢

二、 ER 圖



三、 正規化

(1) User 表格

未正規化（UNF）：包含使用者的姓名、身分與借書資料。

1NF：將重複借書行為分離，只保留單一使用者基本屬性（user_id, name, role）。

2NF：user_id 為主鍵，每個非主鍵欄位（name, role）都完整依賴於 user_id，無部分依賴。

3NF：屬性間無遞移依賴（例如 role 並不依賴 name），符合 3NF。

(2) Book 表格

1NF：每本書一筆紀錄，所有欄位值皆為原子值（不可再分）。

2NF：主鍵為 book_id，其餘欄位（title, author, category, status）都完全依賴 book_id。

3NF：沒有欄位彼此依賴（如 category 與 title 無依賴關係），符合 3NF。

(3) Borrow 表格

1NF：每一次借書紀錄一筆，不含重複群組。

2NF：主鍵為 borrow_id，其餘欄位（user_id, book_id, borrow_date, due_date, return_date）完全依賴主鍵。

3NF：return_date 與 due_date 為獨立欄位，不依賴其他非主鍵欄位，符合 3NF。

(4) Fine 表格

1NF：每筆罰金記錄一筆資料，無重複結構。

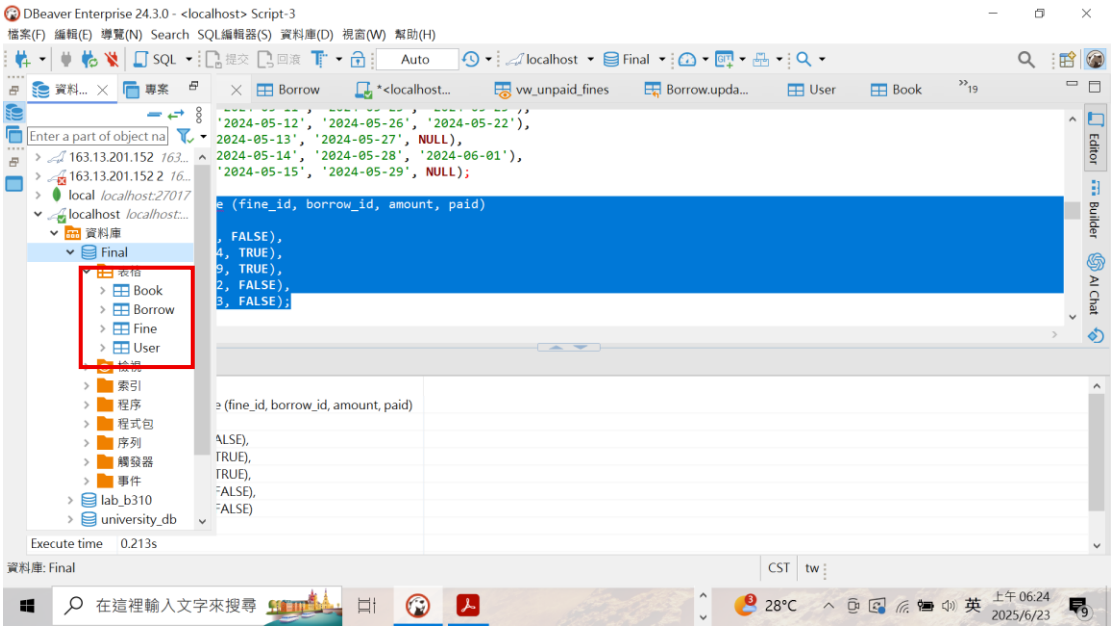
2NF：fine_id 為主鍵，所有欄位都依賴 fine_id。

3NF：amount 與 paid 皆不依賴其他非主鍵欄位，符合 3NF。

四、 資料庫建立與初始資料插入

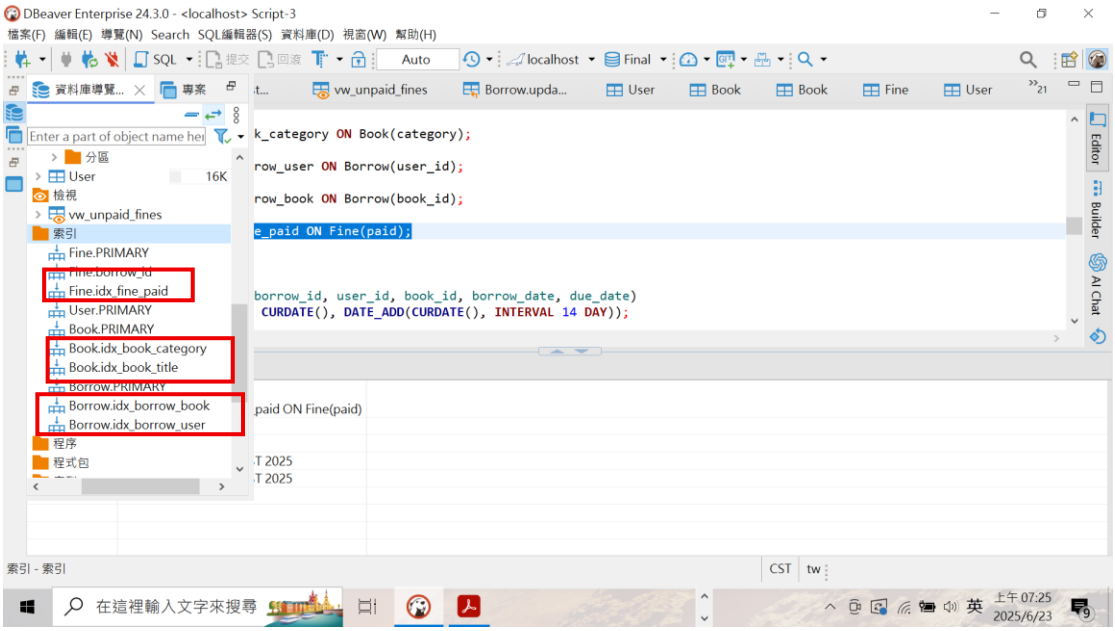
- Book 表存放書本 ID、書名、作者與書籍是否為可借閱狀態
- Borrow 表放借閱編號、借閱書籍的使用者 ID、被借閱的書籍 ID、借閱日、到期日、還書日
- Fine 表放罰款編號、借閱編號、罰款金額、是否繳清罰款
- User 表放使用者 ID、使用者名稱、使用者角色

Book、Borrow、User 表分別插入初始資料 15 筆，Fine 表插入 5 筆。



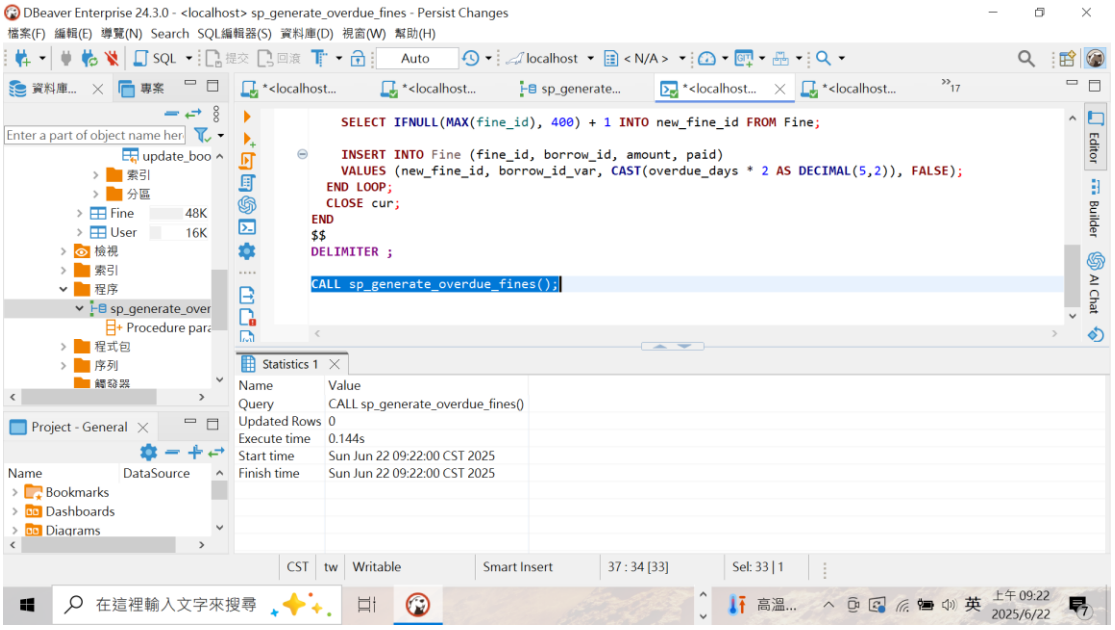
五、 建立索引

- idx_book_title ON Book：加快搜尋書名（模糊或精確比對）。
- idx_book_category ON Book：快速分類查詢圖書
- idx_borrow_user ON Borrow：加快查詢某位使用者的所有借書紀錄
- idx_borrow_book ON Borrow：查詢某本書的所有借閱紀錄
- idx_fine_paid ON Fine：快速查出所有尚未繳清的罰款



六、 建立 Stored Procedure-逾期自動產生罰金

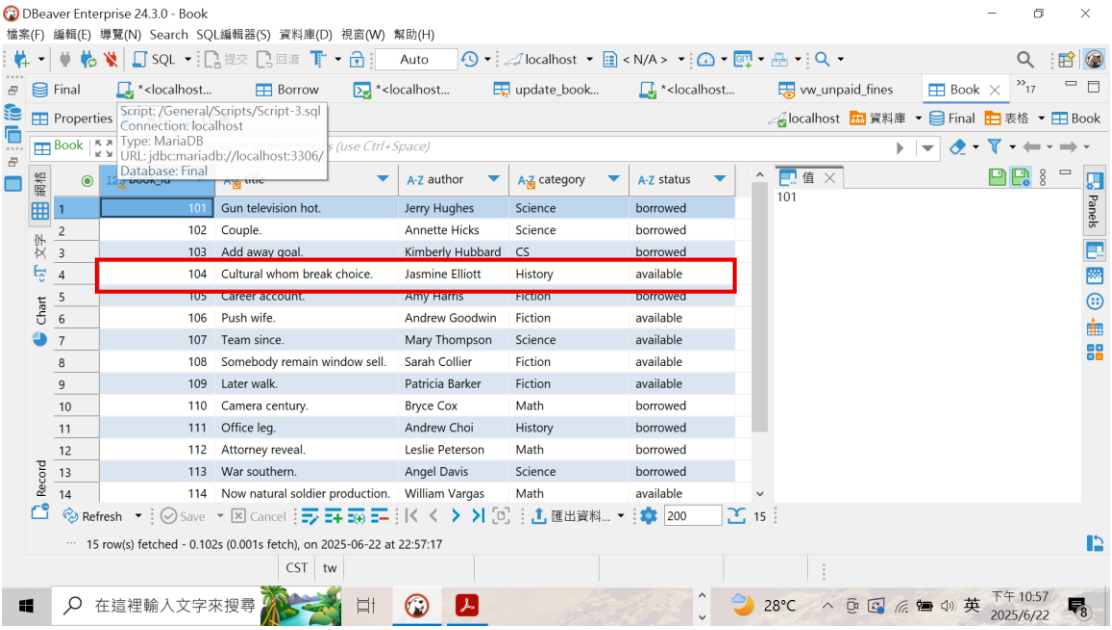
建立 Stored Proceduce-sp_generate_overdue_fines()，可依目前未歸還但逾期的借閱紀錄自動產生罰金，罰金金額會是逾期天數*2 元，以下測試確認可以呼叫

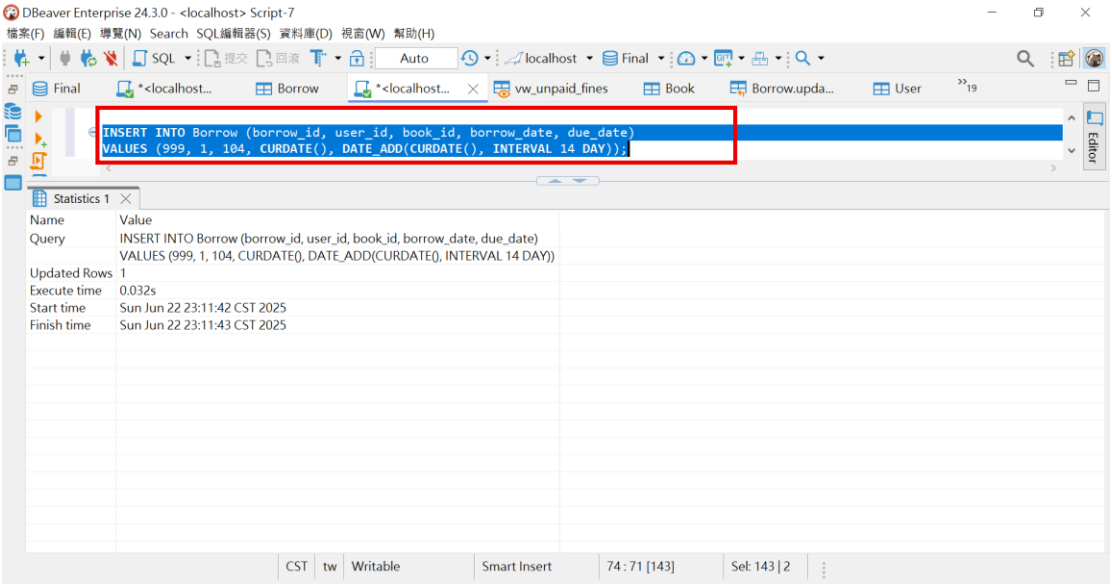


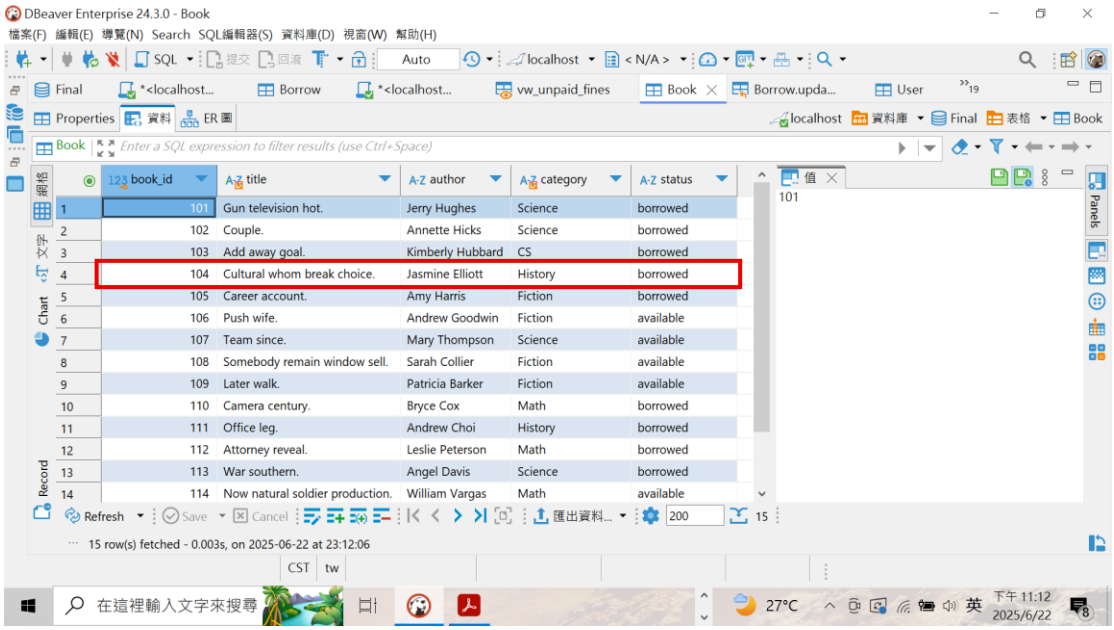
七、 建立 Trigger-借書時自動將書籍狀態設為 "borrowed"

建立 Trigger 的 update_book_status，當書籍被借出時會自動將 Book.status 更新為 borrowed

- 以下測試 Trigger 是否正確執行：確認 book_id 為 104 的書本狀態為 available，讓它被借出，最後確認書本狀態為 borrowed


DBeaver Enterprise 24.3.0 - Book
Script: /General/Scripts/Script-3.sql
Connection: localhost
Type: MariaDB
URL: jdbc:mariadb://localhost:3306/
Database: Final
book_id
title
author
category
status
1 101 Gun television hot. Jerry Hughes Science borrowed
2 102 Couple. Annette Hicks Science borrowed
3 103 Add away goal. Kimberly Hubbard CS borrowed
4 104 Cultural whom break choice. Jasmine Elliott History available
5 105 Career account. Amy Harris Fiction borrowed
6 106 Push wife. Andrew Goodwin Fiction available
7 107 Team since. Mary Thompson Science available
8 108 Somebody remain window sell. Sarah Collier Fiction available
9 109 Later walk. Patricia Barker Fiction available
10 110 Camera century. Bryce Cox Math borrowed
11 111 Office leg. Andrew Choi History borrowed
12 112 Attorney reveal. Leslie Peterson Math borrowed
13 113 War southern. Angel Davis Science borrowed
14 114 Now natural soldier production. William Vargas Math available
101
Refresh
Save
Cancel
匯出資料...
200
15
15 row(s) fetched - 0.102s (0.001s fetch), on 2025-06-22 at 22:57:17
CST
tw
在這裡輸入文字來搜尋
28°C
英
下午 10:57
2025/6/22

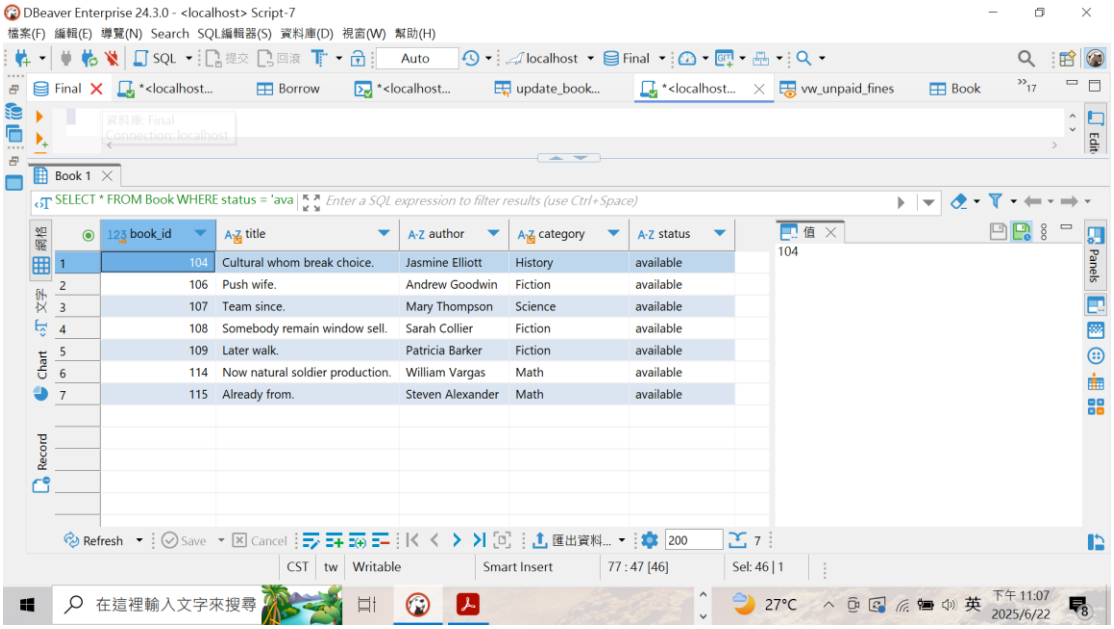

DBeaver Enterprise 24.3.0 - <localhost> Script-7
INSERT INTO Borrow (borrow_id, user_id, book_id, borrow_date, due_date)
VALUES (999, 1, 104, CURDATE(), DATE_ADD(CURDATE(), INTERVAL 14 DAY));
Statistics 1
Name Value
Query INSERT INTO Borrow (borrow_id, user_id, book_id, borrow_date, due_date)
VALUES (999, 1, 104, CURDATE(), DATE_ADD(CURDATE(), INTERVAL 14 DAY))
Updated Rows 1
Execute time 0.032s
Start time Sun Jun 22 23:11:42 CST 2025
Finish time Sun Jun 22 23:11:43 CST 2025
CST
tw
Writable
Smart Insert
74 : 71 [143]
Sel: 143 | 2



八、 功能查詢清單

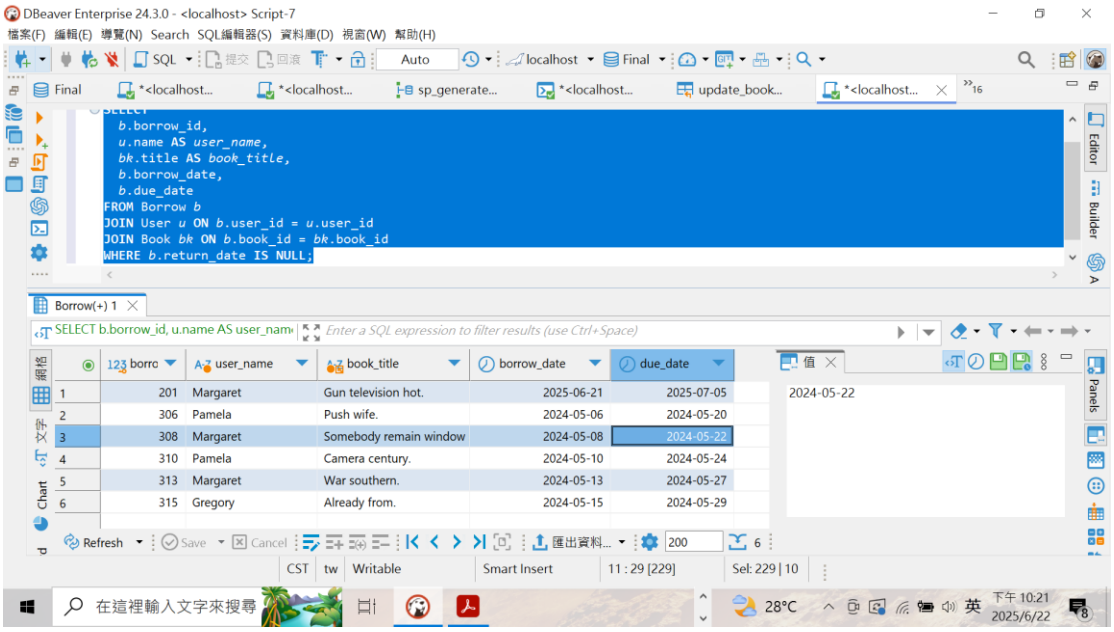
(1) 可借圖書查詢

```
SELECT * FROM Book WHERE status = 'available';
```



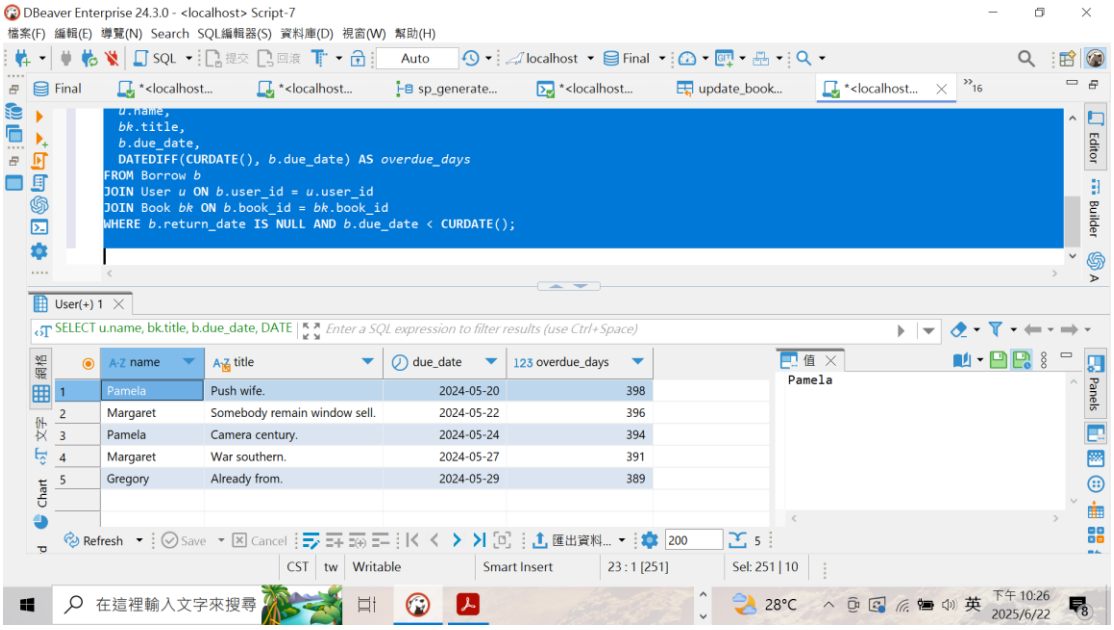
(2) 查詢所有未歸還書籍的借閱紀錄

```
SELECT
    b.borrow_id,
    u.name AS user_name,
    bk.title AS book_title,
    b.borrow_date,
    b.due_date
FROM Borrow b
JOIN User u ON b.user_id = u.user_id
JOIN Book bk ON b.book_id = bk.book_id
WHERE b.return_date IS NULL;
```



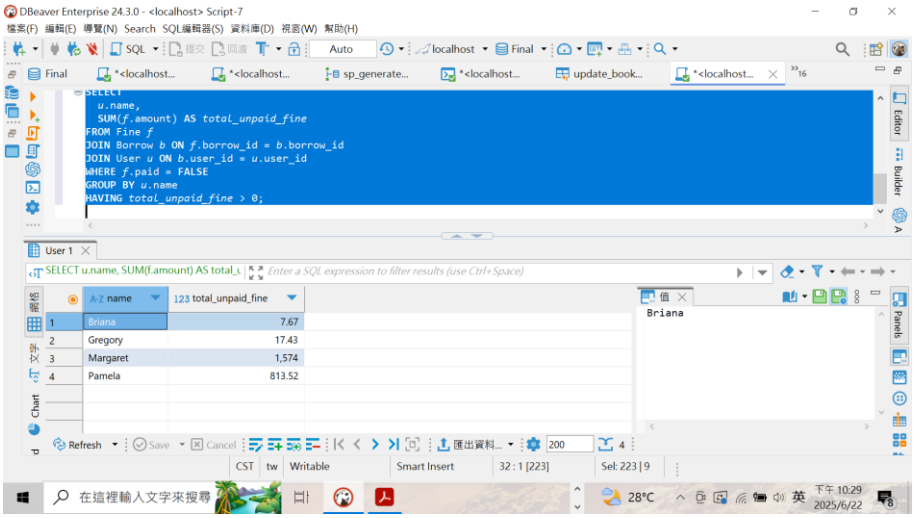
(3) 查詢逾期未還書的使用者名單（含逾期天數）

```
SELECT
    u.name,
    bk.title,
    b.due_date,
    DATEDIFF(CURDATE(), b.due_date) AS overdue_days
FROM Borrow b
JOIN User u ON b.user_id = u.user_id
JOIN Book bk ON b.book_id = bk.book_id
WHERE b.return_date IS NULL AND b.due_date < CURDATE();
```



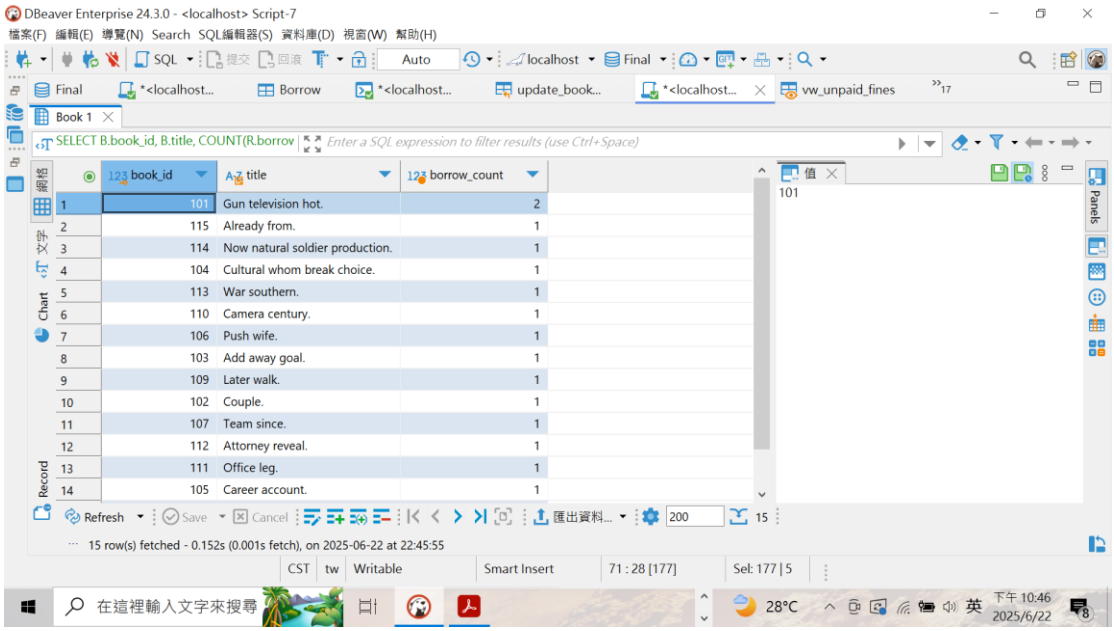
(4) 查詢每位使用者目前應繳總罰金（未繳者）

```
SELECT
    u.name,
    SUM(f.amount) AS total_unpaid_fine
FROM Fine f
JOIN Borrow b ON f.borrow_id = b.borrow_id
JOIN User u ON b.user_id = u.user_id
WHERE f.paid = FALSE
GROUP BY u.name
HAVING total_unpaid_fine > 0;
```



(5) 顯示每本書被借的次數(熱門書籍統計)

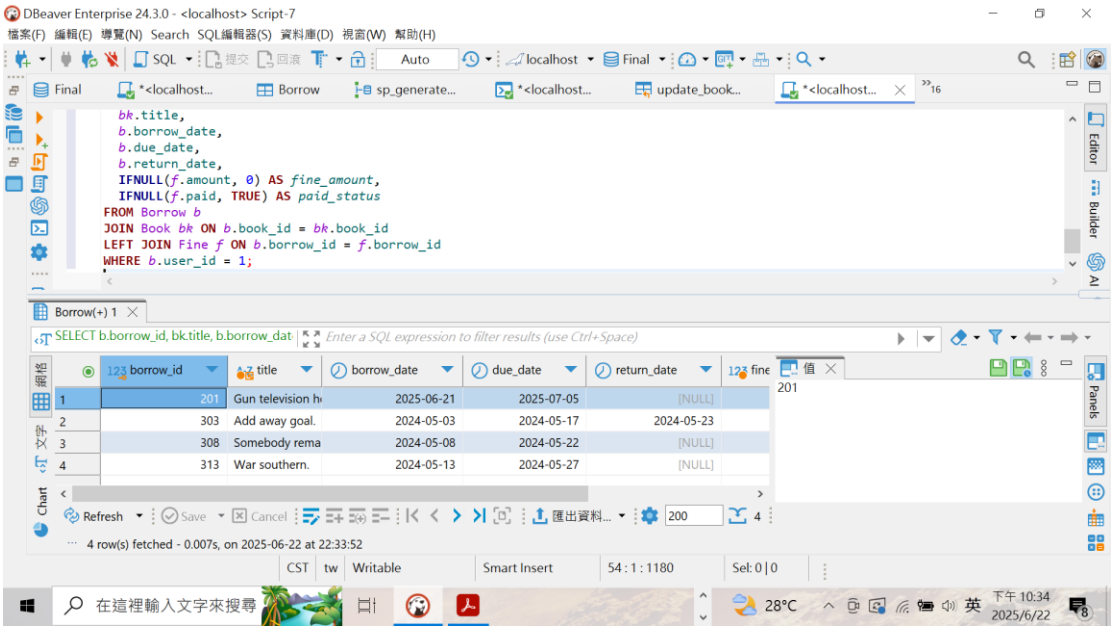
```
SELECT B.book_id, B.title, COUNT(R.borrow_id) AS borrow_count
FROM Book B
LEFT JOIN Borrow R ON B.book_id = R.book_id
GROUP BY B.book_id, B.title
ORDER BY borrow_count DESC;
```



(6) 查詢某使用者借閱與還書紀錄（帶上罰金）

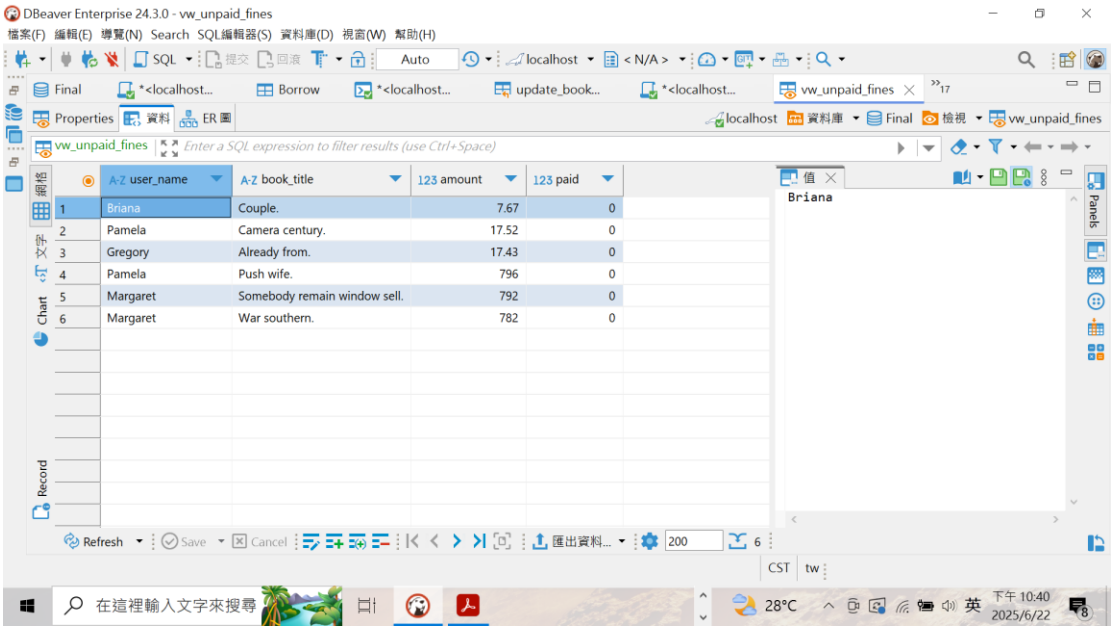
- 查詢 user_id 為 1 的使用者借閱與還書紀錄

```
SELECT
b.borrow_id, bk.title, b.borrow_date, b.due_date, b.return_date,
   IFNULL(f.amount, 0) AS fine_amount,
   IFNULL(f.paid, TRUE) AS paid_status
FROM Borrow b
JOIN Book bk ON b.book_id = bk.book_id
LEFT JOIN Fine f ON b.borrow_id = f.borrow_id
WHERE b.user_id = 1;
```



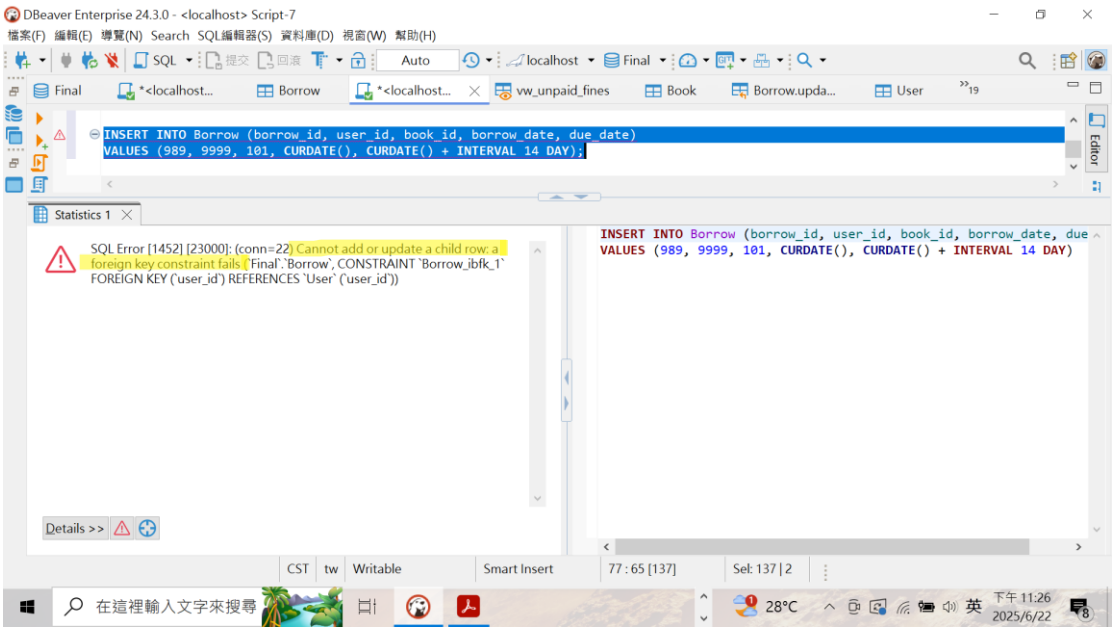
(7) 建立 View：逾期未繳罰金明細

```
CREATE OR REPLACE VIEW vw_unpaid_fines AS
SELECT
    u.name AS user_name,
    bk.title AS book_title,
    f.amount,
    f.paid
FROM Fine f
JOIN Borrow b ON f.borrow_id = b.borrow_id
JOIN User u ON b.user_id = u.user_id
JOIN Book bk ON b.book_id = bk.book_id
WHERE f.paid = FALSE;
```



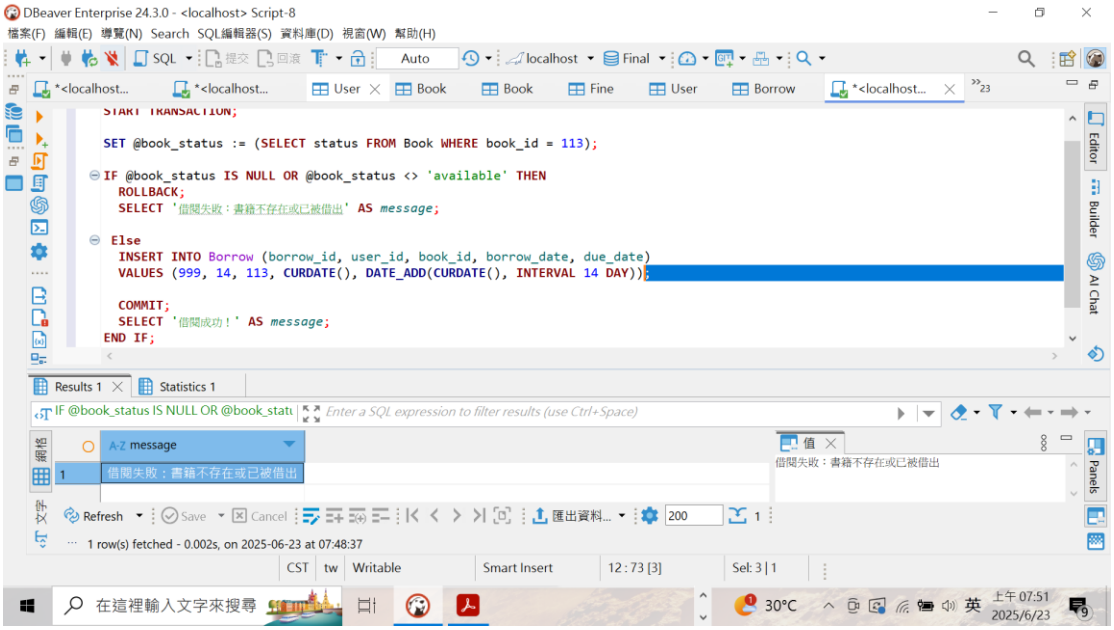
九、 資料一致性測試：外鍵約束與限制條件

插入不存在 user_id 的借書紀錄，測試外鍵是否成功，回報 Cannot add or update a child row: a foreign key constraint fails，外鍵成功防止資料插入



十、　交易處理測試：確保操作一致性

START TRANSACTION 後，確認書本狀態，如果是狀態不等於 available 就 ROLLBACK 並顯示借閱失敗，如果不是就 COMMIT 後顯示借閱成功



十一、　索引效能測試

以 idx_book_title ON Book 測試，先測試沒有索引的查詢執行時間，再創建索引後再執行一次查詢