

程式設計（112-1）作業七一書面報告

以物件導向程式設計管理圖書資訊實例

b12705055 張佑丞

November 2023

1 簡介

假設今天我們要開發一套系統，用以管理圖書館中的圖書資訊，為此我們需要建立一套完善的圖書資訊管理系統，這個系統會存儲大量的圖書資訊，包含哪位使用者在何時借閱了哪本書籍，這不僅能大大減低圖書館員的負擔，也能夠大幅減少過往時常出現的人工疏失，更提供了我們許多使用者與其決策內容供我們分析，未來也能依此作為圖書資訊推播的參考依據。

2 理由

由於以下原因，我認為使用「物件導向程式設計」（object-oriented programming）會是一個不錯的解決方法。

1. 封裝 (Encapsulation)：在 OOP 裡面，資料與函式會被綁在同一個物件當中，這讓程式碼的可閱讀性增加以及方便後續維護管理。舉例來說，在這個圖書資訊管理系統中可以有一個名為 Book 的物件，而他包含了 title, author 等等的資訊，也能建立如 checkOut() 和 return() 的函式。這能將所有跟 Book 有關的資料、功能全部統一管理，讓不同開發者間能更容易讀懂其程式碼，也方便未來程式碼的維護。
2. 模組化 (Modularity)：OOP 提倡模組化的方式管理系統，也就是說不同物件能夠獨立地開發以及測試，彼此並不會互相影響。這不但能夠增加程式開發的效率也能提高程式碼的可重複利用性。

3. 隱藏原始資料 (Data hiding): OOP 允許隱藏原始資料, 也就是在內部的物件能夠保證不被外界直接地存取到, 這能夠避免資料不被不小心地更動, 也可以增進程式碼的穩健性 (robustness)。

而相比之下, 程序程式設計最大的缺點就是缺乏統整性, 在程序程式設計中, 資料和函式是分開的, 也就是隨著程式的複雜程度增加, 開發者會越來越難去追蹤這些資訊怎麼串接在一起的。而緊接在後的就是資料外洩的可能性以及低可重複利用性的程式碼。