****

**OBJECT ORIENTED PROGRAMMING 1**

RATIONALE

MINI LIBRARY MANAGEMENT SYSTEM

In this project, I used three main python data structures to manage the library system:

* A dictionary for books
* A list of dictionaries for members
* A tuple for genres.

Here is why each choice makes sense:

1. **Books** stored in a dictionary

Each book is saved using it ISBN(a unique book ID) as the key. This makes it very fast and easy to find, add, update, or remove book. For example, If I need to check how many copies of book are available, I will just look it up by ISBN, no need to search through a long list.

1. **Members** stored in a list of dictionaries

Each member (like a library user) is represented as a small dictionary containing their ID, name, email, and a list of books they have borrowed. All members are kept in a single list. This works well because:

* It is simple to add new members.
* I can easily loop through the list to find a specific member by their ID.
* It clearly shows which books each person has borrowed.

1. **Genres** stored in a tuple

Valid book genres(like “fiction”,”sci-fi”, or “Biography”) are stored in a tuple. A tuple is like a list that cannot be changed after it’s created. This is useful because:

* It prevents accidental changes(like adding a mispelled genre).
* It makes sure only approved genres are used when adding or updating books.
* Checksing if a genre is valid is quick and safe.

**CONCLUSION**

These choices keep the system organized, easy to understand, and reliable. They follow the assignment requirements while using python’s built-in features effectively without overcomplicating the code. The result is a mini library system that works smoothly for basic tasks like borrowing, returning, and managing books and members.