|  |
| --- |
| CWP1: Project Library Management |
|  |
| Himanshu Verma  (M00885750) |

# **Project Overview:**

### **objective**: A small library needs a system to track the details of their available books and members. They provided you with the data source CSV file.

# **Key Components:**

### **Enum and Struct for Book Types:** I have built a struct BookTypeMapping to map strings to these enum values, and an enum class BookType for different book genres. This method of handling book categories is effective.

**Date Class:**

### Date-related operations, such as assigning due dates and adjusting dates that surpass allowable boundaries, are managed by a custom Date class. The scheduling of book due dates for borrowed materials is greatly aided by this class.

**Person, Member, and Librarian Classes:**

### To demonstrate inheritance, the **Person class** acts as the foundation for the Member and Librarian classes.

### Derived from Person, the **Member class** manages features exclusive to library users, such as checking out and returning books.

### The **Librarian class**, which is similarly developed from Person, has more extensive duties such as managing books and members and lending and returning books.

**Book Class:**

### Represents the specifics and workings of a library book, such as the procedures for borrowing and returning books.

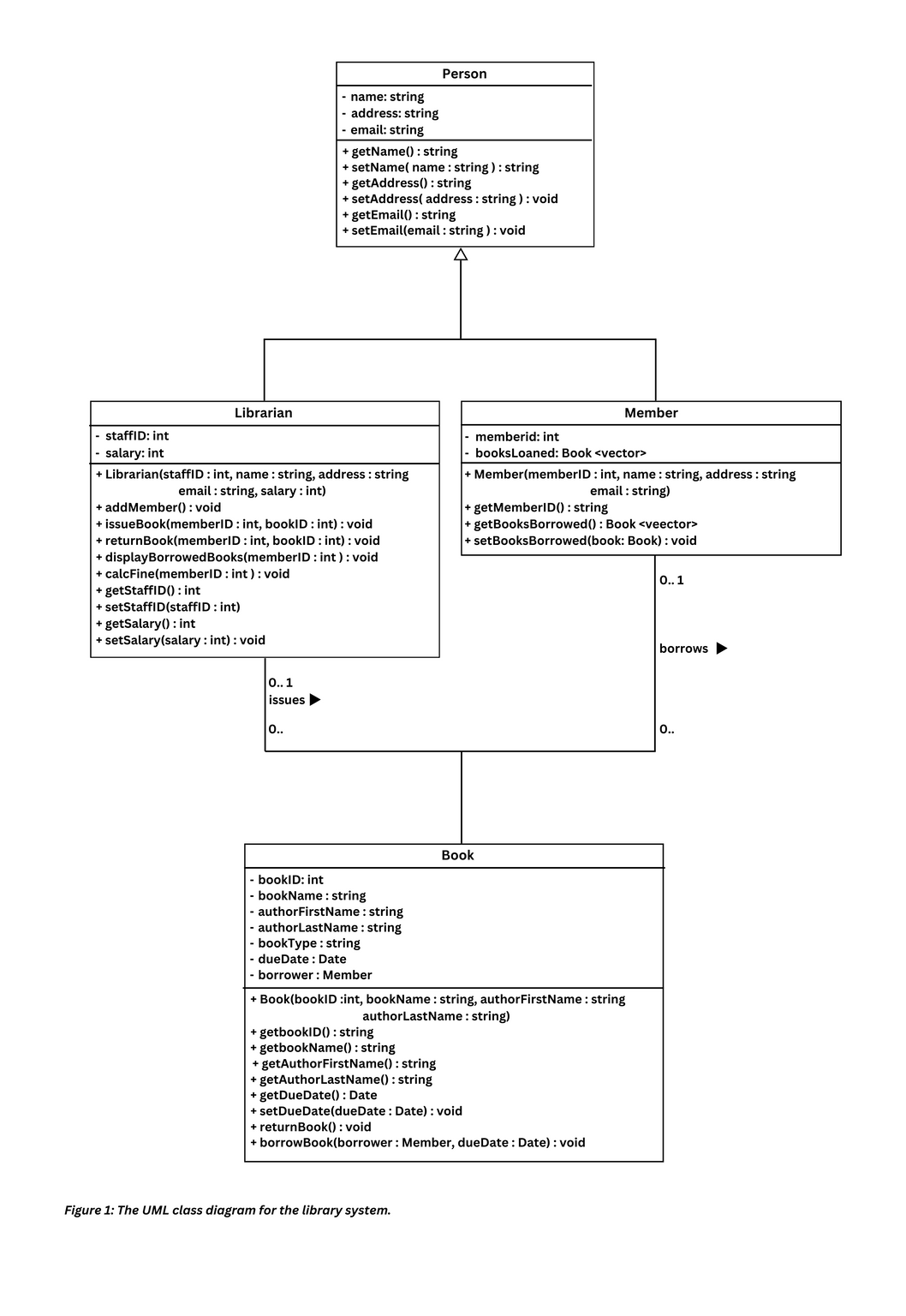
**File Reading for Book Initialization:**

### In order to demonstrate file handling in C++, the librarian class reads from a CSV file to initialise the book collection.

**Interactive Menu-driven Interface:**

The primary feature facilitates user involvement with the system by offering an interactive menu-driven interface for library administration operations.

# **UML Diagram:**



The system should include the following functionality:

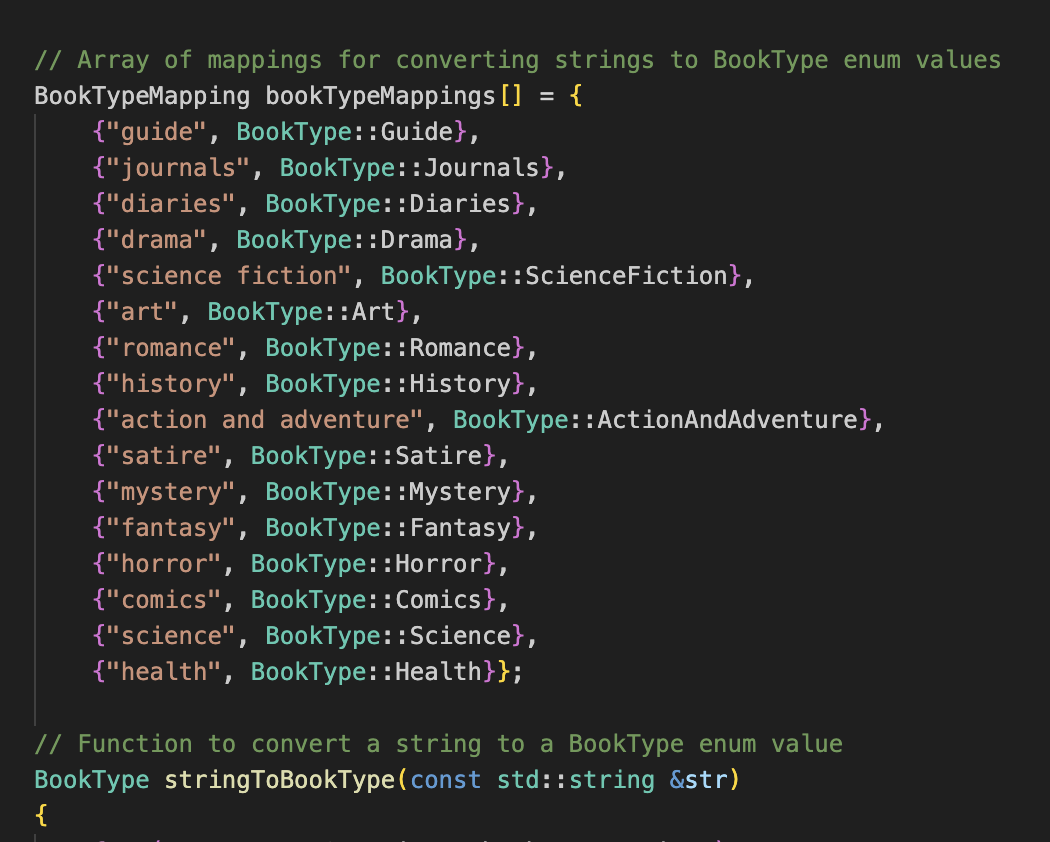
**Add a member** – the librarian should be able to create a new member and display the new member’s details directly following the creation of the member.

**Issue a book to a member** – the librarian should be able to issue a book to an individual member with a valid due date from the date of issue (3 days).

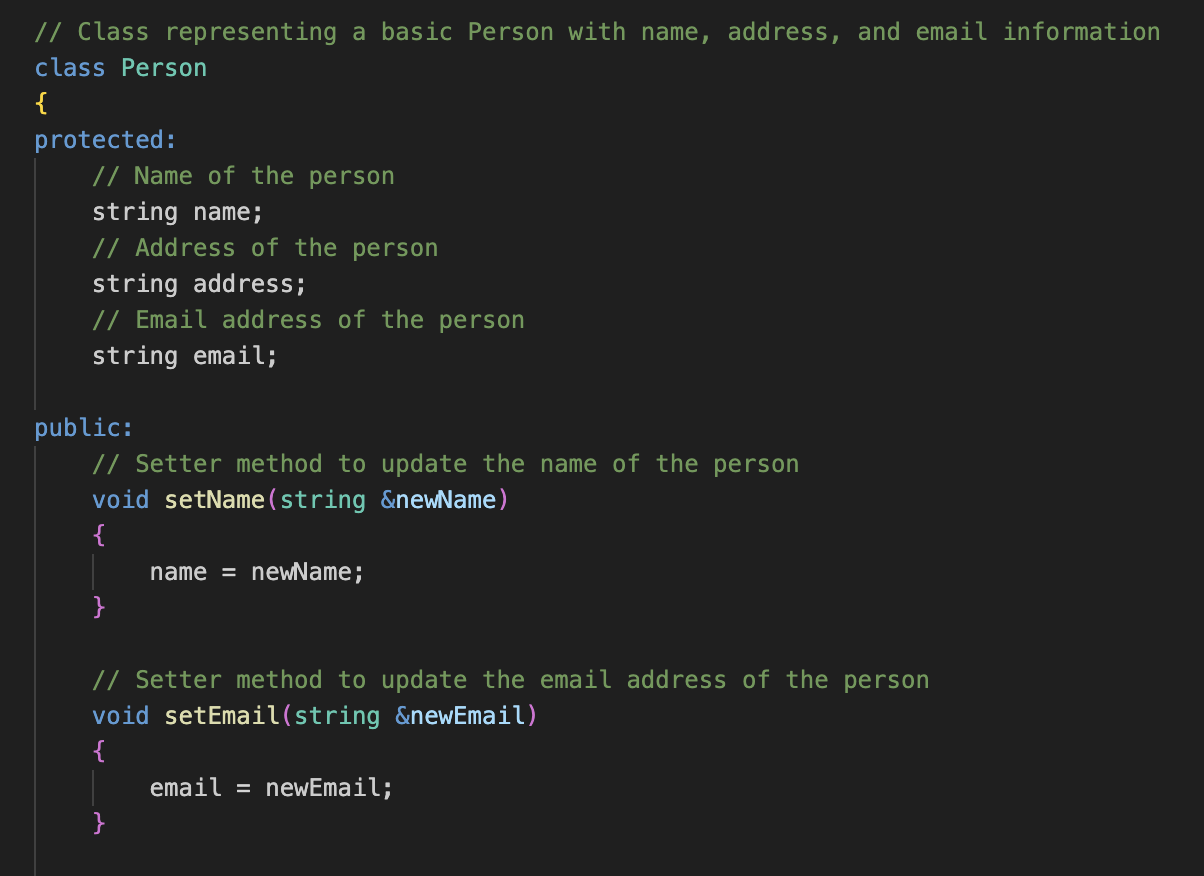
**Return a book** – the librarian should be able to return a book from an individual member.

**Display all books borrowed by any individual member** – the librarian should be able to display all books borrowed by an individual member

**Calculate a fine for any individual member for overdue book(s)** – upon the return of a book, if the book’s due date has expired, a fine should be calculated based on a rate of £1 per day overdue.

Array of mappings for converting strings to BookType enum values

Class representing a basic Person with name, address, and email information

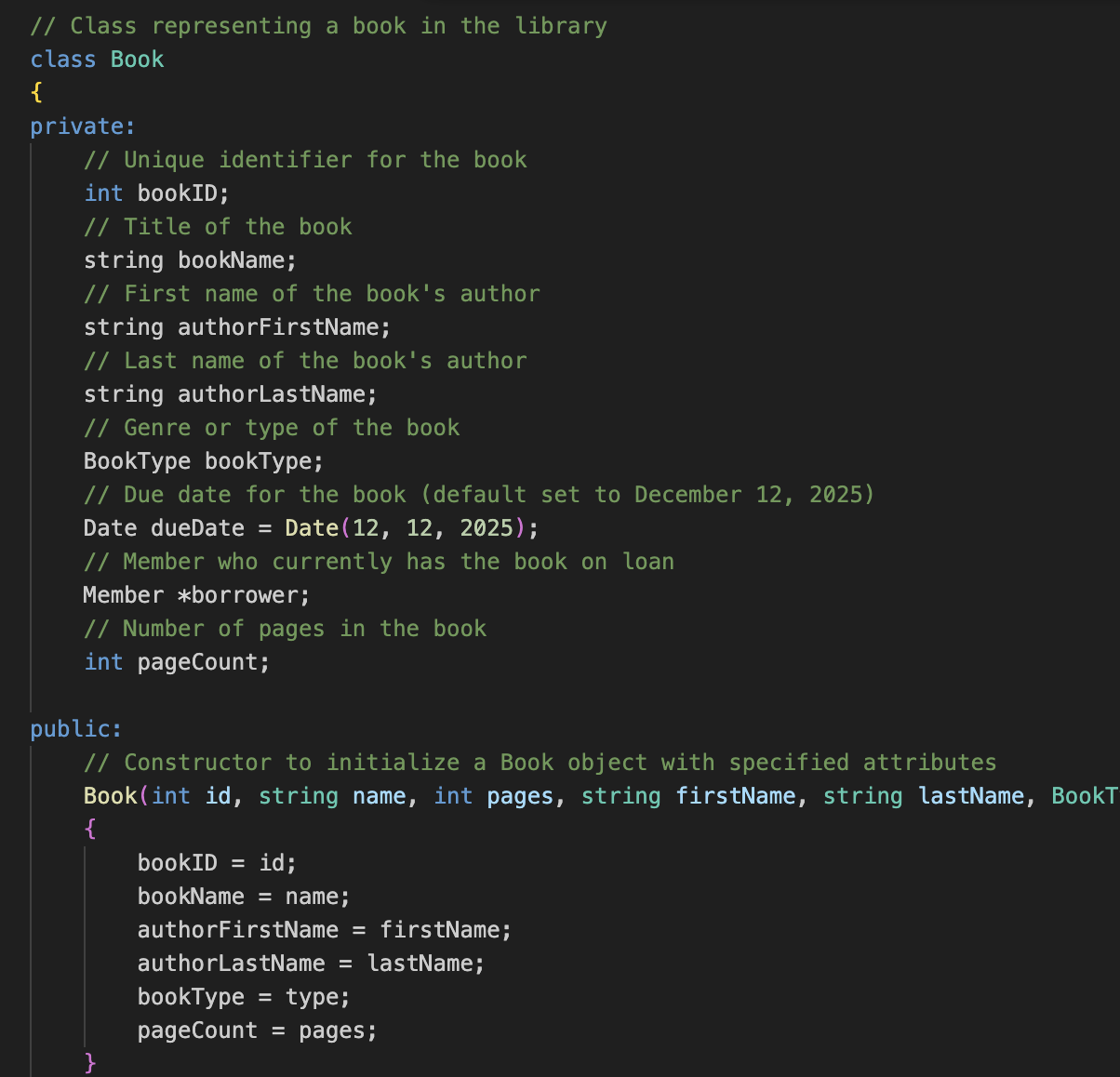


**Person**:

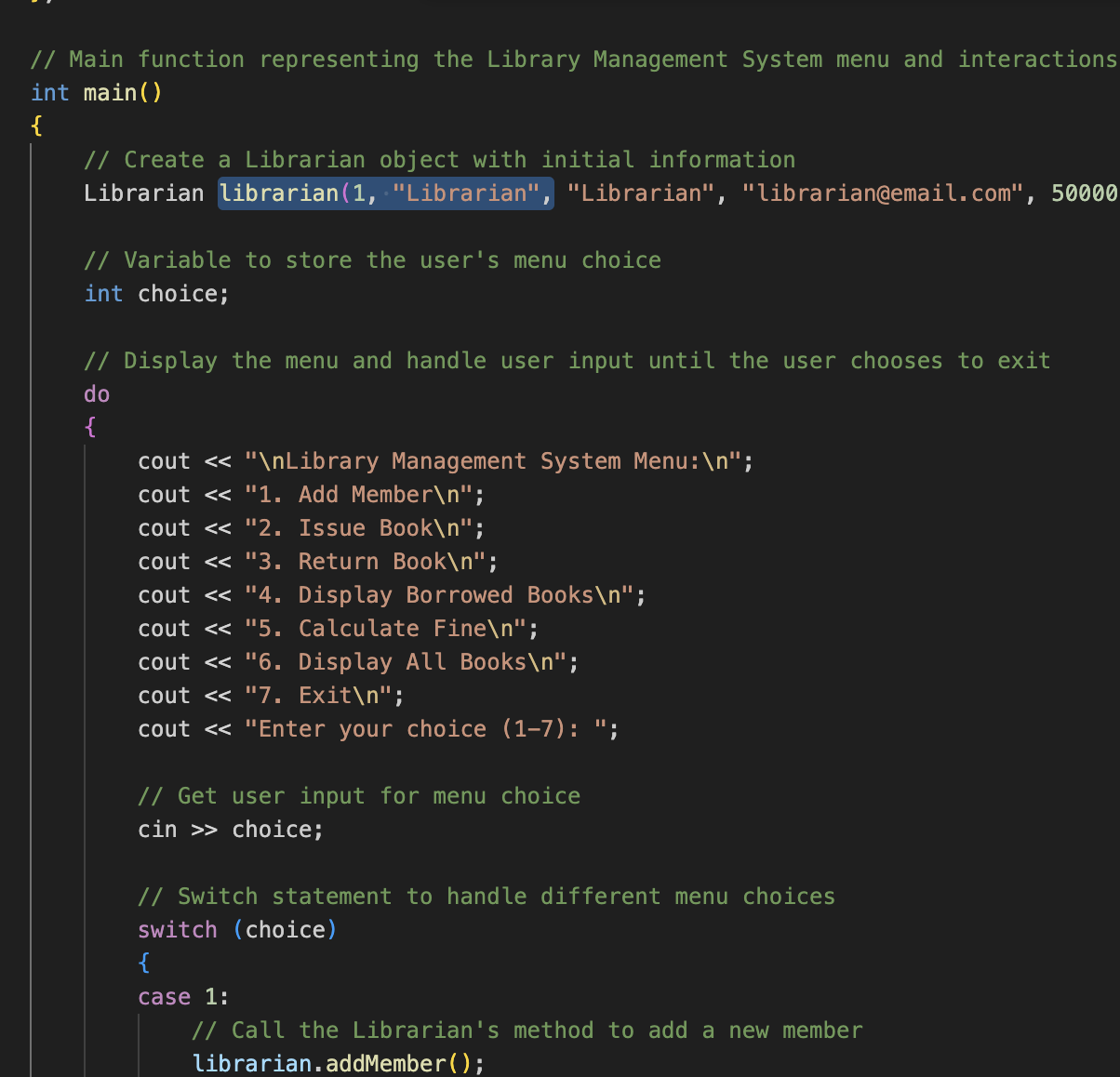
This is a base class that has email, address, and name attributes.

It provides ways to set and get these properties.

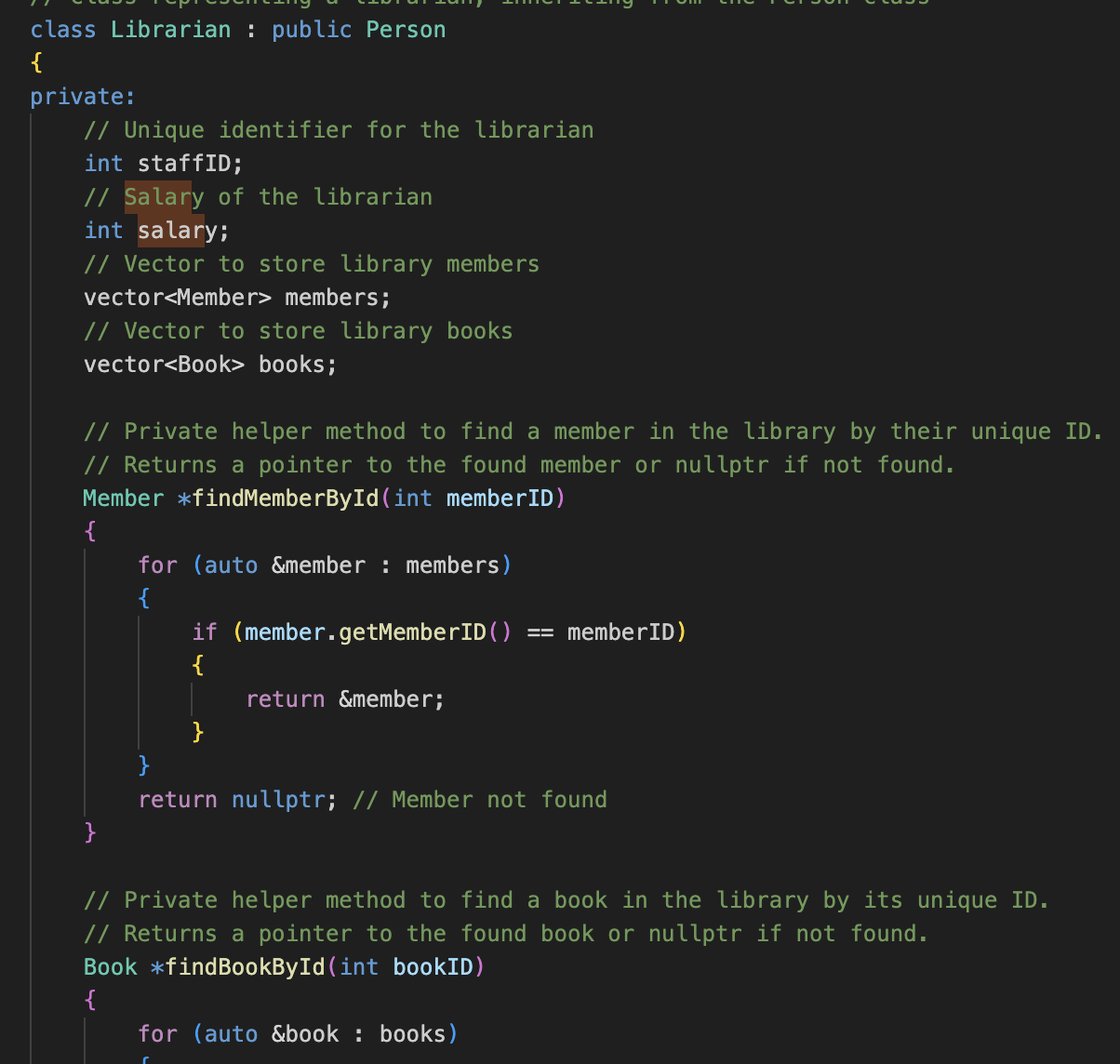
Class representing a book in the library & Constructor for Book class



// Main function representing the Library Management System menu and interactions.



// Class representing a librarian, inheriting from the Person class



Librarian Class: This class inherits from Person and has additional attributes:

staffID: A unique identifier for a librarian.

salary: The salary of the librarian