Python Web – Mid Project

You need to create a program for managing a library. You have the following classes :

Class : Book

Data Members:

- - author (string)
- - title (string)
- - num_of_pages (int)

Class : Shelf

Data Members:

- - books (a list of Maximum 5 Book objects)
- - is_shelf_full (boolean)

Functions

- - addBook – receives a Book object and add it to the books list in the first available. If there is no more space , a proper message will be printed. If the shelf become full (with 5 Book objects), than the “is_shelf_full” will be set to True. Otherwise, will be set to False.
- - replace_books – receives 2 numbers between 1 and 5 and replace between the Books in these locations. If one of the location is empty, a proper message will be printed.
- - order_books – order the books by their num_of_pages in ascending order.

Class : Reader

Data Members:

- - id (int)
- - name (string)
- - books (a list of dictionaries with titles of the books he read and the dates he took them from the library)

Functions

- read_book – receives a book title and adds it + the current date to the books list

Class : Library

Data Members:

- - shelves (a list of 3 Shelf objects)
- - readers (a list of Reader objects)

Functions

- - is_there_place_for_new_book - Returns a Boolean indicates if there is a place for inserting a new book to the library
- - add_new_book - receives a new Book object and add it to the first Shelf with a free space
- - delete_book – receives a book title and delete the Book object from the library.
- - change_locations – receives 2 books titles, and replace between these 2 Books objects (their locations in the shelves).

- change_locations_in_same_shelf – receives a shelf number, and books locations , and replace between these 2 Books objects.
- - order_books – order all books in each shelf by their num_of_pages
- - register_reader – receives a new reader name and id and add it to the readers list.
- - remove_reader – receives a new reader name and removes it from the readers list.
- - reader_read_book – receives a book title and a reader name and add this book title to the reader's books list
- - search_by_author – receives an author name and returns all books titles this author wrote

The Program

The program manages ONLY 1 Library object.

The program starts with pre-defined 2 Books object on EACH shelf. These books are pulled from a proper collection in a MongoDB data base (The collection stores 6 Books records – for 3 Shelves)

This library management system is accessible only to registered library employees. To start working with system, the employee must login with his username and email login details. These details will be verified against the REST API at https://jsonplaceholder.typicode.com/users

If there's an employee with that username and email he will be logged in and will get the following menu (in infinite loop) :

- - **"For adding a book - Press 1".**
- - **"For deleting a book - Press 2".**
- - **"For changing books locations - Press 3".**
- - **"For registering a new reader - Press 4".**
- - **"For removing a reader - Press 5".**
- - **"For searching books by author – Press 6."**
- - **"For reading a book by a reader – Press 7."**
- - **"For ordering all books – Press 8."**
- - **"For saving all data – Press 9".**
- - **"For loading data – Press 10".**
- - **"For exit – Press 11".**

If the user press 1 – The program will ask the user for all new book data and adds it to the library by calling "add_new_book" of the Library object.

If the user press 2 – The program will ask the user for the book title and remove it from the library by calling "delete_book" of the Library object.

If the user press 3 – The program will ask the user for the 2 books titles he want to replace, and replace them by calling "replace_locations" of the Library object.

If the user press 4 – The program will ask the user for the new reader name will add him to the readers list by calling "register_reader" of the Library object.

If the user press 5 – The program will ask the user for the reader name remove him from the readers list by calling "remove_reader" of the Library object.

If the user press 6 – The program will ask the user for the author name print all author's books titles by calling "search_by_author" of the Library object.

If the user press 7 – The program will ask the user for reader id and the book title and add this book to the reader's books list by calling "read_book" of the proper Reader Object.

If the user press 8 – The program order all books by calling "order_books" of the Library object.

If the user press 9 – The program ask the user for a file name and saves all library data in a JSON file on that given name as follows :

```
{
    "shelves" : [
        {
            "is_shelve_full" : false,
            "books" : [
                {
                    "author" : "J.K.Rolling",
                    "title" : "HarryPotter",
                    "num_of_pages" : 100
                },
                {
                   // More books data
                }
            ]
        },
        {
            // more shelves data...
        }

    ],
    "readers" : [
        {
            "id" : 111,
            "books" : [{"title" : "Harry Potter" , "date" : "10/03/2012"},
                       {"title" : "IT" , "date" : "12/01/2009"}]
        },
        {
            //More readers data
        }
    ]
}
```

If the user press 10 – The program ask the user for a file name and loads all Library data from that JSON file with the given name

If the user press 11 – The program ends.

How to deliver the project ?

If you like your project to be code reviewed, you need to upload the code to any repository that you want (google drive, git,aws…)
and send a link to it to devprojects2000@gmail.com
The subject need to be in the following format :

[name] – [course name] – [project name]

For example : Avi Cohen – Full Stack 33 – Python Web mid project