

**NAME:**

SYED MUHAMMAD MUBASHIR RIZVI

**ROLL NO:**

CS-20071

**SUBMITTED TO:**

MS. IBSHAR ISHRAT

# PROBLEM STATEMENT:

Implement a library management system having the following functionalities:

* **Add/Remove/Edit book:** To add, remove or modify a book or book item.
* **Search catalogue:** To search books by title, author, subject or publication date.
* **Register new account/cancel membership:** To add a new member or cancel the membership of an existing member.
* **Check-out book:** To borrow a book from the library.
* **Reserve book:** To reserve a book which is not currently available.
* **Renew a book:** To re-borrow an already checked-out book.
* **Return a book:** To return a book to the library which was issued to a member

1. Implement and use any sorting algorithm to sort the books on the basis of any attribute (e.g. book ID, number of books, author, title, etc.).
2. Prepare the complete software (in Python)
3. Insert at least 15 records in the system
4. Implement the feature to display the whole collection of books.

**NOTE**

1. OEL is on individual basis
2. Last date of submission: SUN 6th Feb, 2022
3. Deliverables: 1. Program code 2. List of books 3. Report
4. Total marks: 10 (see the rubrics below)

|  |  |  |  |
| --- | --- | --- | --- |
| **S.N.** | **Rubrics** | **Max Marks** | **Marks Obtained** |
| 1 | *Implement* sorting algorithm | 2 |  |
| 2 | *Prepare* the complete software | 4 |  |
| 3 | *Prepare* the final report | 4 |  |

# APPROACH AND ASSUMPTIONS:

I have used the Bubble sort algorithm to sort the books on the basis of their title name and to store all the data i.e. of the user accounts, the list of books, and the information regarding whether the book is borrowed, reserved, or renewed is done with the help of filing system of Python. To update the records in all the files, I opened the file in write mode and then written all the records including the edited ones or excluding the ones to be removed. The assumption made by me is that all of the functions and the working is being done by an admin who is already registered in the management and when some function like borrowing is to be done, only then the customer is required to log in.

# FILES USED:

1. **List of Books:** This file contains all the books with the following format

Book Name-Author-Quantity-Floor on which it is located

1. **Users:** This file contains the user or customer accounts with the following format

Email/Username-Password

1. **Books Information:** This file contains the information of book renewal, check out or reservation in the following format

Email/Username-Function-Book Name

# FUNCTIONS IMPLEMENTED:

## LOGIN:

The user is asked to enter the email/username and password and it is then checked with all the accounts in the Users.txt file. If the entered information exists in the file, the function returns true and also returns the credentials used to log in, otherwise false is returned.

## SIGNUP:

The user is asked to enter the email/username and password and afterwards, the email is checked with all the accounts in the Users.txt file, if a duplicate is found, the function prints a message and returns, otherwise it returns true and the account is registered.

## ADD BOOK:

First of all, the records in the books file are fetched and displayed, after which book name, title, quantity, floor are asked and then they are entered in the file which contains all the books.

## REMOVE BOOK:

Firstly, all the records are fetched and displayed after which the book name and quantity is required for removal. The record is then completely removed if the quantity becomes zero, otherwise the quantity is then subtracted. To update it in the file, the updated list of the books is written again by opening the List of all Books file in write mode.

## EDIT BOOK:

Firstly, all the records are fetched and displayed after which the user can select the book which he/she wants to edit. The user has the choice if he/she wants to edit the book name, author, quantity or the floor. The updated book is then written with all the other books in the file by opening the file in write mode.

## SEARCH BOOK:

The user can search on the basis of the author name or the book name, and then all the records are fetched from the List of all Books file, after which they are checked if they contain the name of the book or author. If any or more matches are found, they are returned in the form of a list.

## SORT BOOKS:

To sort the books, the books are all fetched from the relevant file and then the algorithm Bubble Sort is used to sort the books with respect to their title name i.e. the books are sorted alphabetically. I did not update the file with the sorted books, this function just returns a list of books which are sorted, if I sorted the books in the file, this function would only be required to be ran one time, however I used this function to display the sorted books to the user only.

## RENEW BOOK:

The user is first asked to login, if the login is successful and the account exists, then 2 things are checked with the previous information of the book. Firstly, it is checked that if the same user has logged in as the one who borrowed the book previously, and secondly, it is checked if it is the same book as before. Then the Books Information file is updated with the information that the user renewed the book.

## CHECK OUT:

The user is first asked to login, and after the login is successful, the list of all the books are fetched and displayed to the user. The user can then select any book and he/she can only borrow 1 book at a time. The quantity is then subtracted and updated in the List of all Books file, and also the Books Information is updated with the information that user borrowed a book.

## RETURN BOOK:

The user is first asked to login, and after the login is successful, all the Books Information is fetched and it is checked if the user had previously borrowed (check out) a book or renewed one, only then the user is allowed to return the book. After this, the List of all Books file is updated with the information that the book has been returned i.e. the quantity is added 1 and the Books Information file is also updated after removing the record of the user checking out or renewing the book.

## RESERVE BOOK:

The user is asked to login and after the login is successful, the user is asked to enter the name of the book he or she wants to reserve. This information is then added into the Books Information file.

# FLOW OF THE PROGRAM/DRIVER CODE:

In the driver code, the user has the following options:

* Add a Book
* Remove a Book
* Edit a Book
* Search a Book
* View all Books
* Check-Out or Borrow a Book
* Renew a Book
* Reserve a Book
* Return a Book
* Login
* Signup
* Cancel Membership
* Exit

Now, the first three options do not require any customer login because all the functions are controlled by an admin who is already logged in. The user can select any of the options which calls their respective functions and displays the respective message. The options continue to come over and over again until the user selects Exit.

## LOGIN OUTPUT:

## 

## SIGNUP OUTPUT:

## 

## ADD BOOK OUTPUT:

## 

## REMOVE BOOK OUTPUT: EDIT BOOK OUTPUT:

## CHECK OUT BOOK OUTPUT: RENEW BOOK OUTPUT:

## RETURN BOOK OUTPUT: VIEW BOOKS OUTPUT:

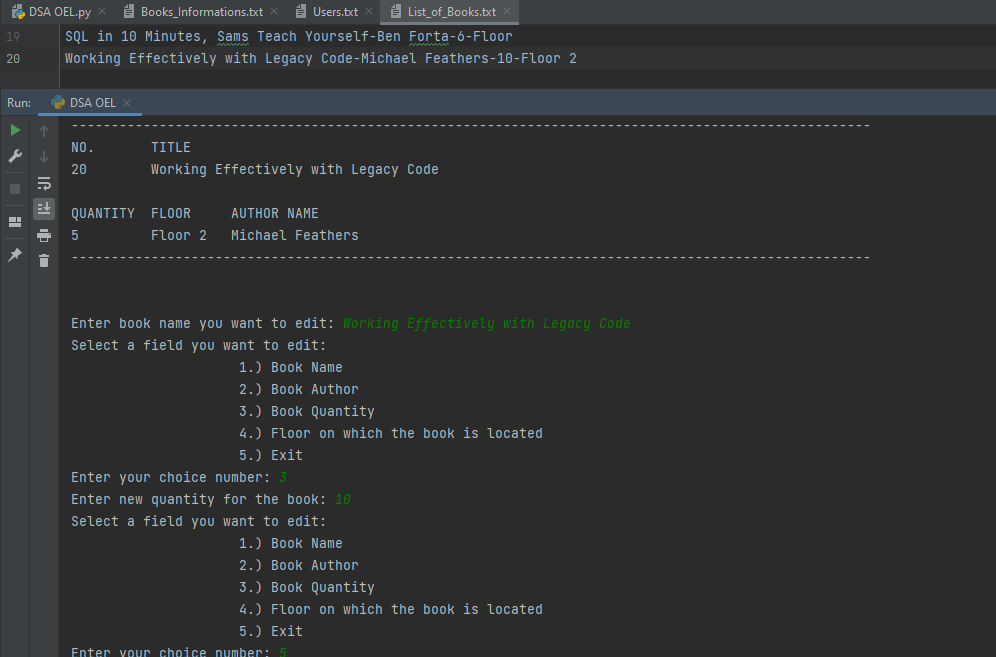
## 

## REMOVE BOOK OUTPUT:

## 

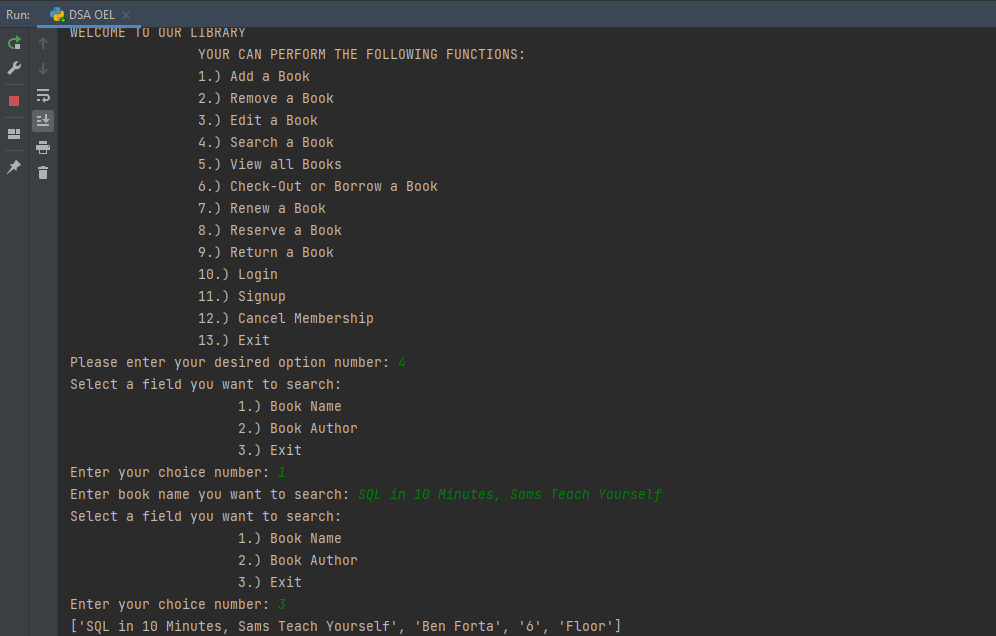
## 

## EDIT BOOK OUTPUT:



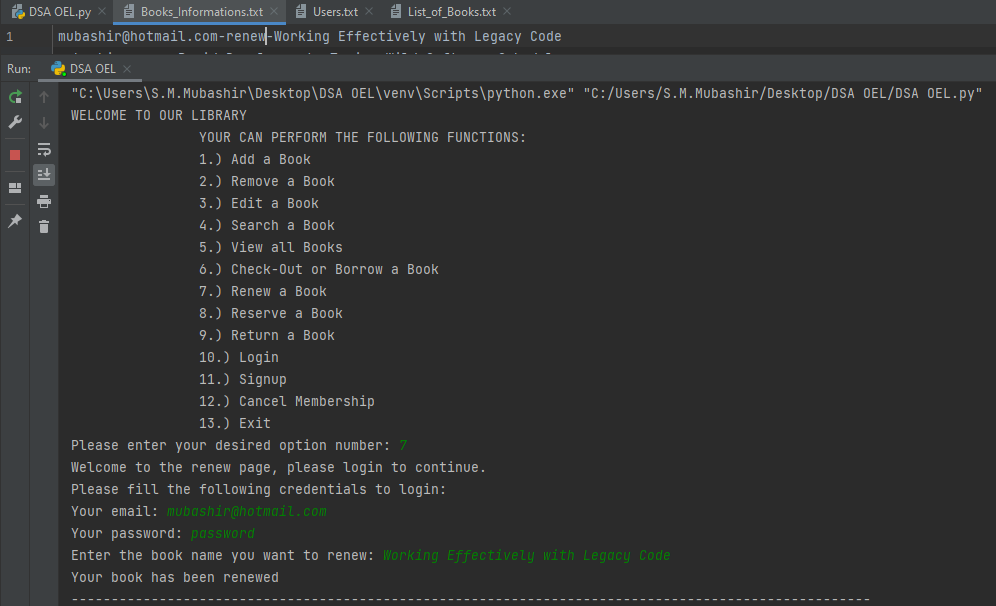
## CANCEL MEMBERSHIP OUTPUT:

## SEARCH BOOK OUTPUT:



## CHECK OUT BOOK OUTPUT:

## RENEW BOOK OUTPUT:



## RETURN BOOK OUTPUT:

## VIEW ALL BOOKS OUTPUT:

