

**BATCH 2020-21**

|  |  |
| --- | --- |
| **NAME** | **ABDUL RAFFAY** |
| **ROLL #** | **CS-094** |
| **CLASS/SEC** | **S.E - C** |
| **COURSE NAME** | **DATA STRUCTURES AND ALGORITHMS** |
| **COURSE CODE** | **CS-218** |
| **SUBMITTED TO** | **MS IBSHAR ISHRAT** |

**DEPARTMENT OF COMPUTER & INFO SYSTEMS ENGINEERING**

**NED UNIVERSITY OF ENGINEERING AND TECHNOLOGY**

**PREFACE.**

This report describes all the core features of the project, the implementation analysis of the project, code snippets of the key features of the app, snapshots of the test runs of the app.

**TABLE OF CONTENTS.**

|  |  |
| --- | --- |
| 1. Overview of the Project | Page no. (3-4) |
| 1. Implementation Analysis | Page no. (4-9) |
| 1. Searching Mechanism Snapshots | Page no. (10-11) |
| 1. Guide To Use The Application | Page no.(12) |

* **OVERVIEW OF THE PROJECT:**

**. 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 reborrow an already checked-out book.

• Return a book: To return a book to the library which was issued to a member

**. INTRODUCTION OF THE PROJECT:-**

This project has been built in order to solve the above problem statement. It has been built purely with Python Programming Language which is a high level language and known widely because of its diversity. In this Library Management System, there are two entities which are The Admin and The User.

The Admin or Librarian has the following options, he/she can:

1. Add a Book.
2. Remove a Book.
3. Update a Book.
4. Delete a Book.
5. See All the Books.
6. Search a Book by its Name, Author, Subject and Publication Date.
7. Sort the Books in an Alphabetical Order.
8. See All the Registered Users.
9. Cancel Membership or Delete Account of a User.

The User or Layman has the following options, he/she can:

1. See All the Books in the Library.
2. Borrow a Book.
3. See Borrowed Books.
4. Reserve a non-available Book.
5. See Reserved Books.
6. Search a Book by its Name, Author, Subject and Publication Date.
7. Return Back a Borrowed Book.
8. Renew an Already Borrowed Book.
9. Delete his/her account.

* **IMPLEMENTATION ANALYSIS:**

This software is completely built with Python. The whole software has been coded by following the Object Oriented Programming Paradigm in Python.

**. Classes Description:**

In this software, there are four classes:

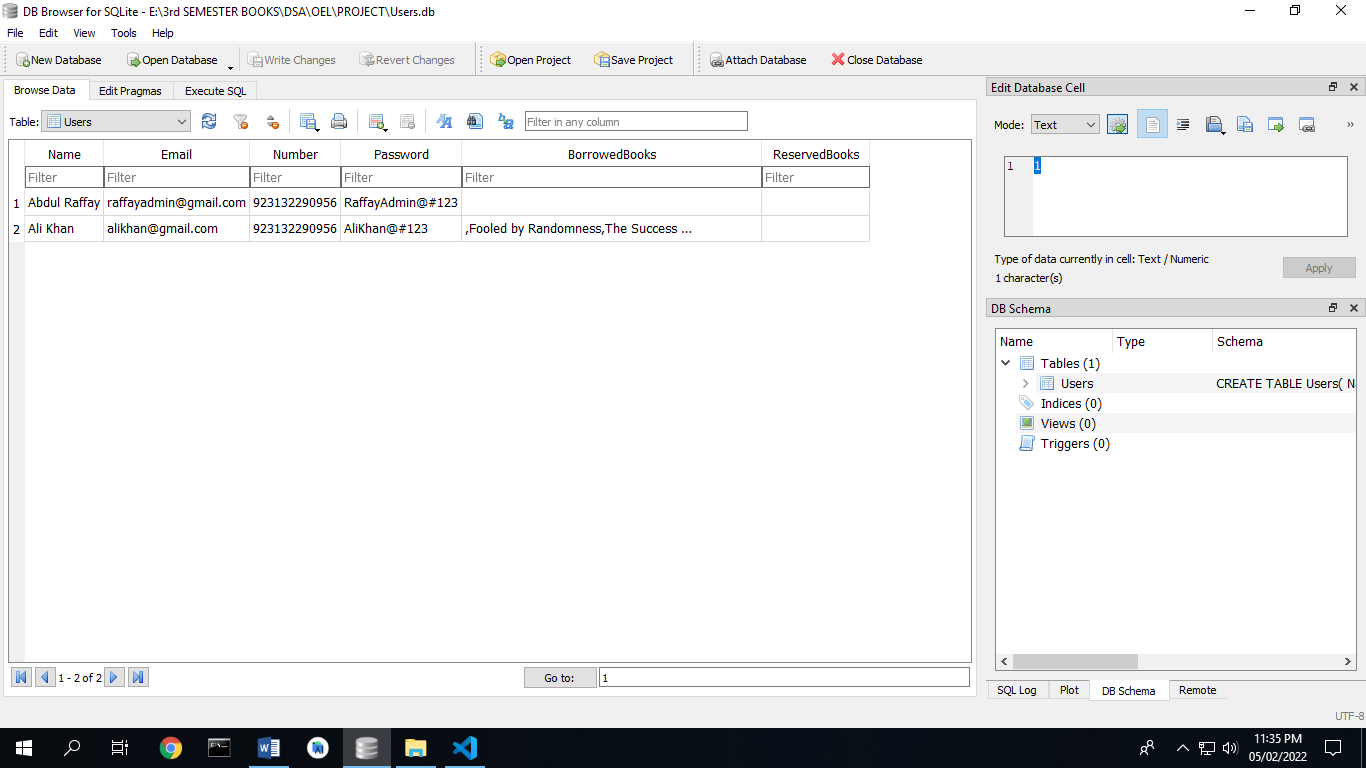
1. **Class User** which deals with all of the Stuff Related to User Credentials and Authentication.
2. **Class Book** which deals with all of the stuff related to the Books in the Library.
3. **Class Algorithm** which deals all the Algorithms used in the development.
4. **Class Main** which controls the whole flow of the Application. It will have objects from the classes **Book, User, and Algorithm** in order to utilize their methods.

**. Database Tables Description:**

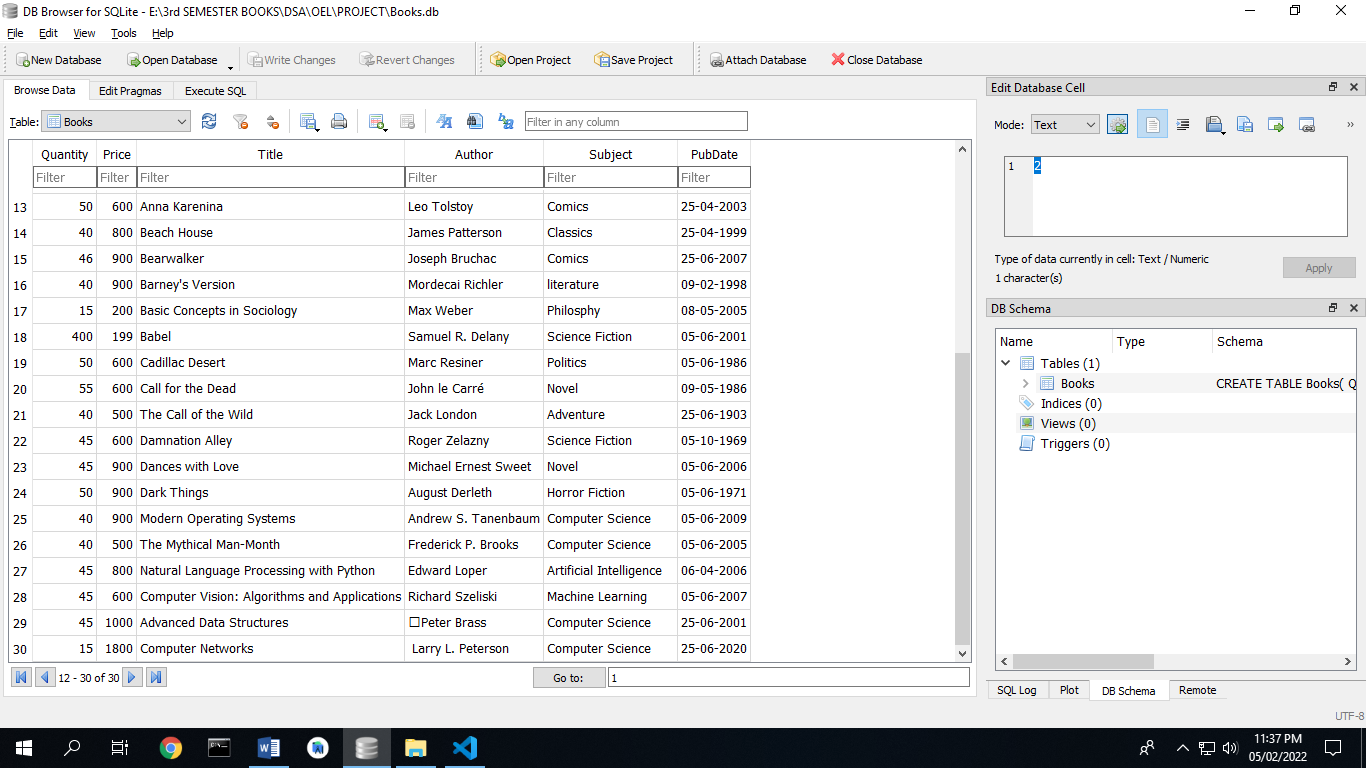
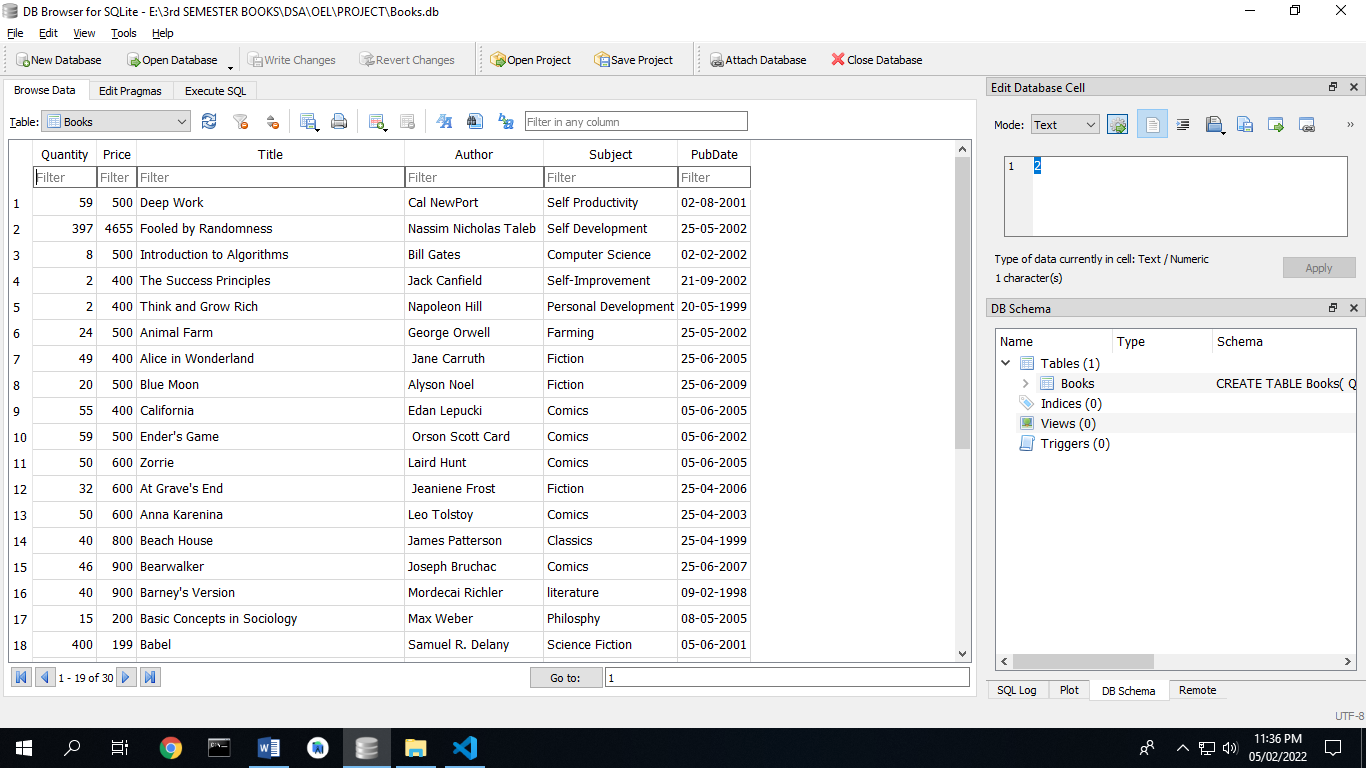
All the data handling like the CRUD (Create, Read, Update, Delete) Operations has been implemented using **sqlite3** Database in python. Mainly there are two tables as follows:

1. Table 1: **User.db** which saves all the Credentials of the user and lists of books which he or she has borrowed or reserved.
2. Table 2: **Books.db** which saves all the information of a book like the title, author name, quantity, price, subject, publication date

Below are attached the screenshots of the Tables **Users.db** and **Books.db.**

1. **Users.db:** 

Currently there are two users in the Table **Users.db**. The very first record is the admin account and afterwards there are the users accounts.

1. **Books.db:** 

Currently there are 30 books in the Table **Books.db** each book having the following fields:

1. Quantity.
2. Price.
3. Title.
4. Author Name.
5. Subject.
6. Publication Date.

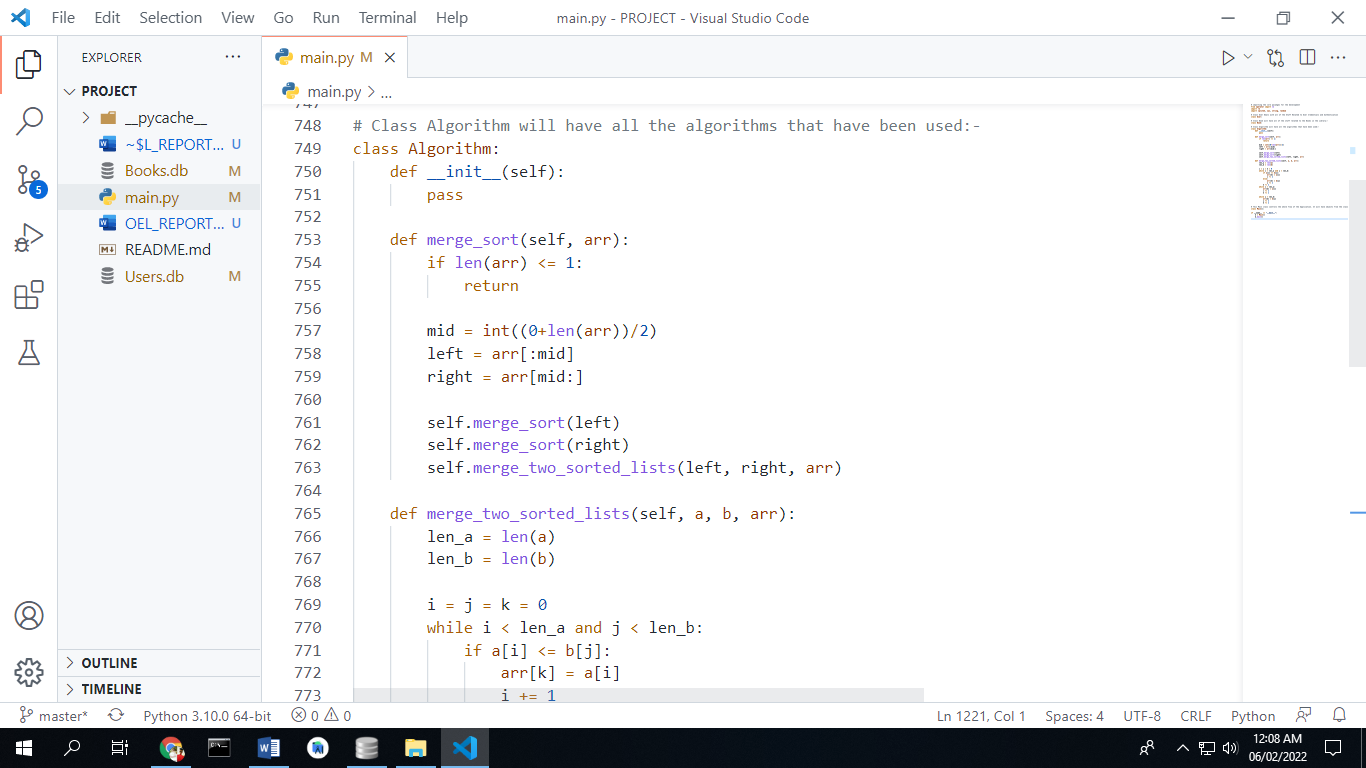
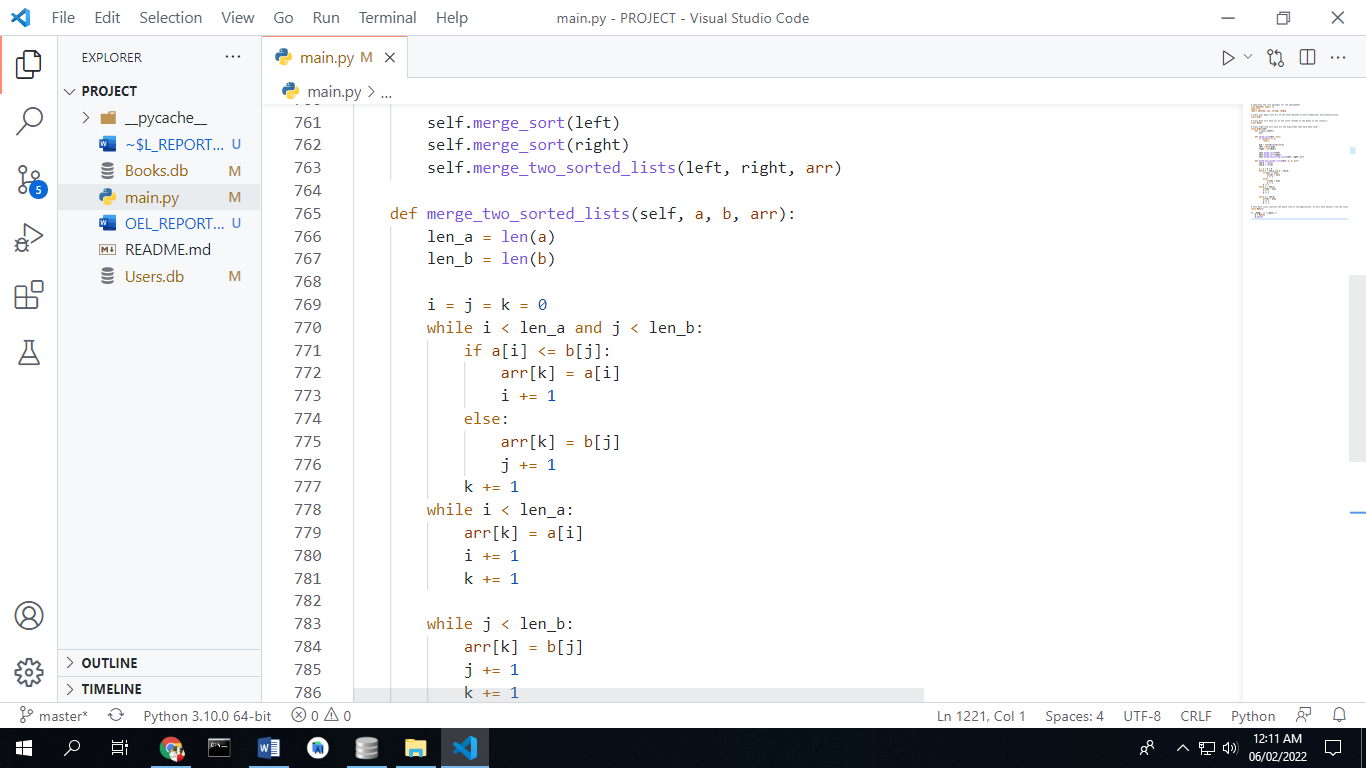
**. Application of Sorting Algorithm to Sort the list of Books:-**

In this program, list of books has been sorted in an alphabetical order by an algorithm known as **“Merge Sort”.** I have used merge sort rather than bubble sort or quick sort because of the following complexity analysis of the three sorting algorithms:

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Big O Complexity in**  **Best Case.** | **Big O Complexity in**  **Average Case.** | **Big O Complexity in**  **Worst Case.** |
| 1. Merge Sort | O(**nlog(n)**) | O(**nlog(n)**) | O(**nlog(n)**) |
| 1. Quick Sort | O(**nlog(n)**) | O(**nlog(n)**) | O(**n²**) |
| 1. Bubble Sort | O(**n²**) | O(**n²**) | O(**n**) |

As can seen from the table above merge sort is the most efficient algorithm for sorting than the other two sorting algorithms because it has a big O Complexity of **O(nlog(n))** in all the three cases that are the best, average and worst case. Below are attached the screenshots of the sorting algorithm that has been implemented in the software and the results after the application of the merge sort. In the code merge sort algorithm has been implemented inside the **Class Algorithm.**

**. Merge Sort Algorithm:**

**. Output:**  

**. Functionality of Borrow and Reserve Book for the user:-**

When the user borrow any book from the library then that book is added to his or her **BorrowedBooks** list and when the user returns that book to the library again then that book is removed from the User’s **BorrowedBooks** list and has been added to the records of the library. When any book is currently not available and if the user makes an attempt to borrow that book then the user will be asked to add that non-available book to his or her **ReservedBooks** list so that as soon as the admin or librarian adds that book, the book will automatically been added to the user’s **BorrowedBooks** list.

**. Functionality of Showing all the Books:-**

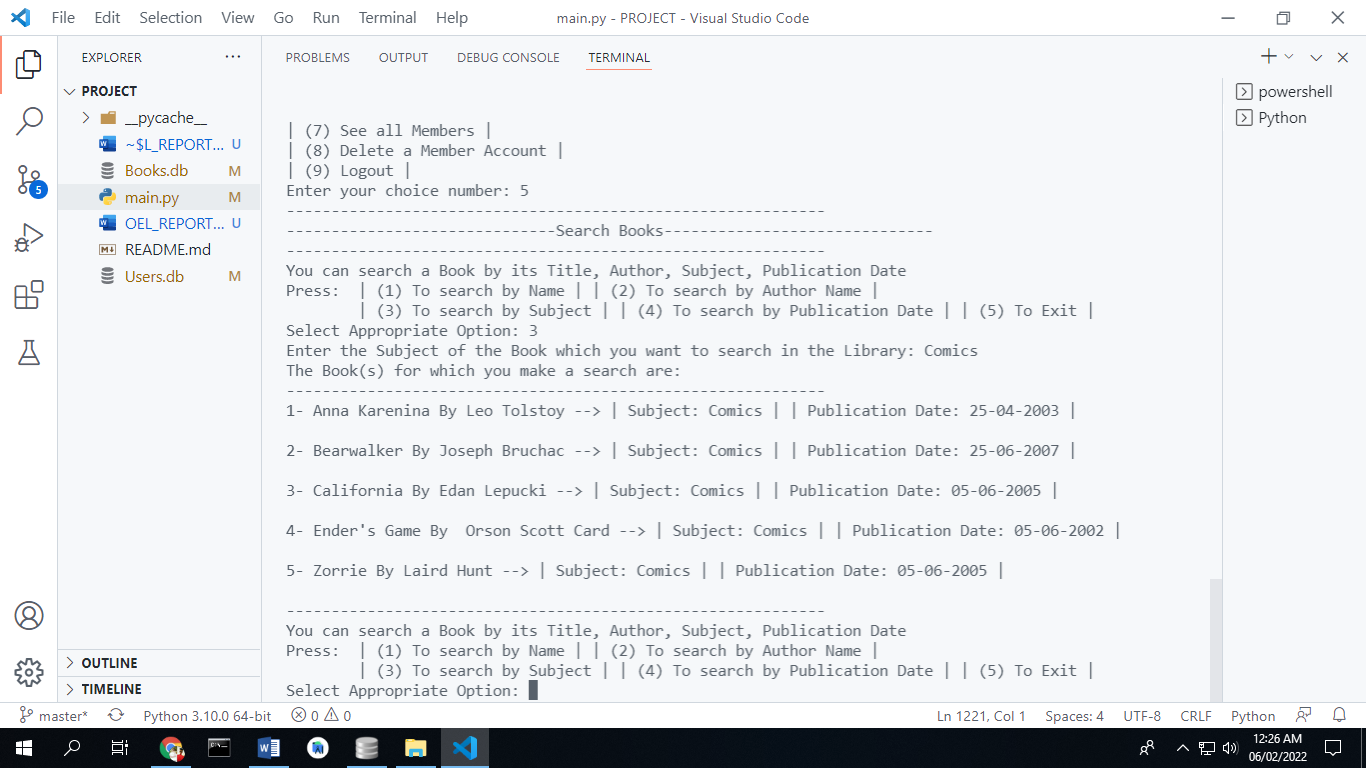
User can see all the books of the libraray, below are attached the screen shots of it:

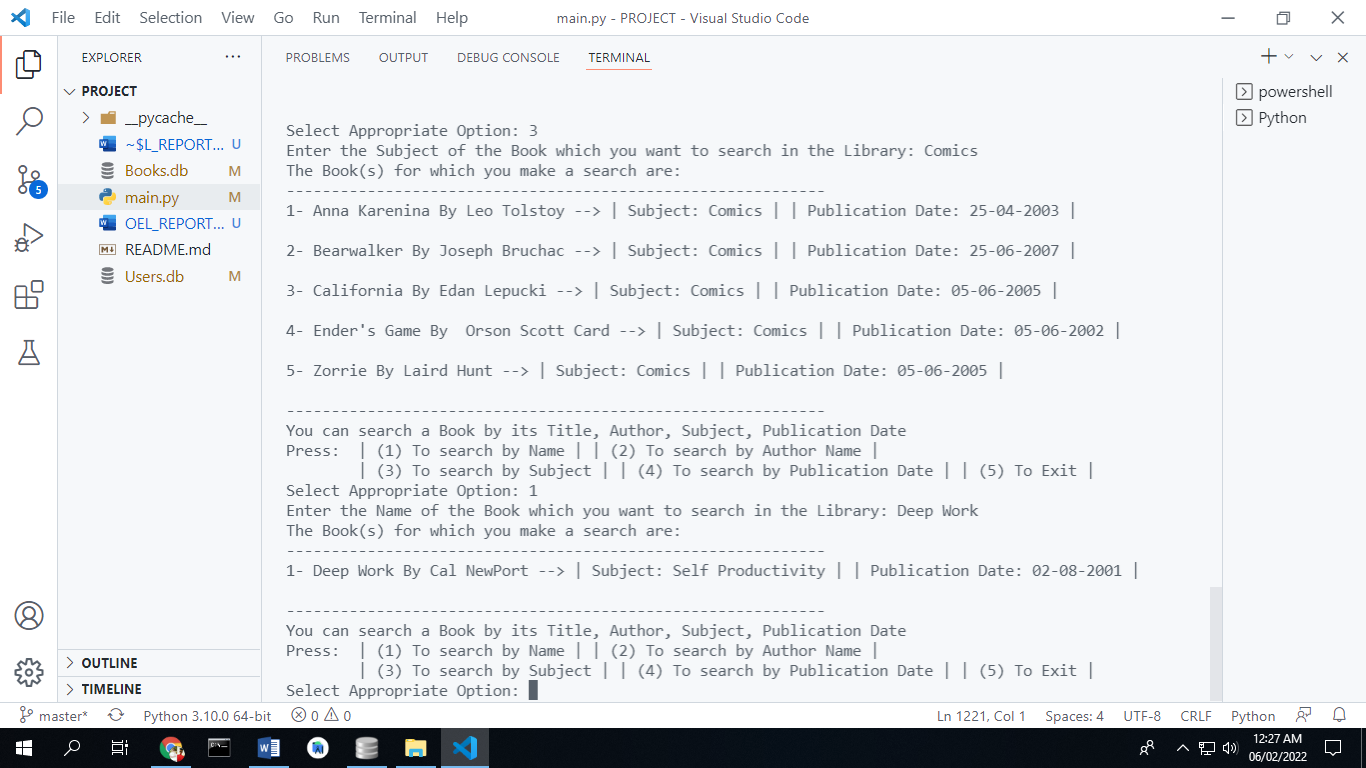


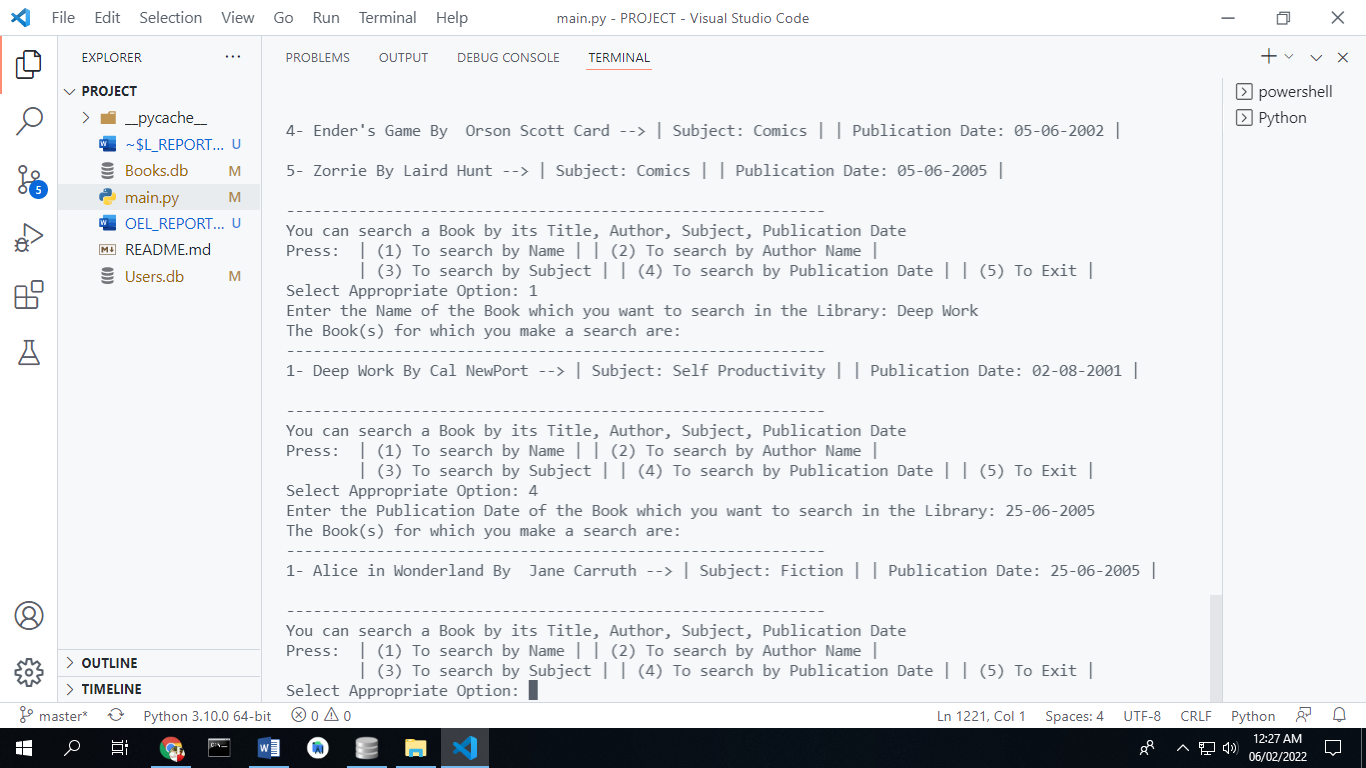
* **Searching Mechanism Snapshots:**

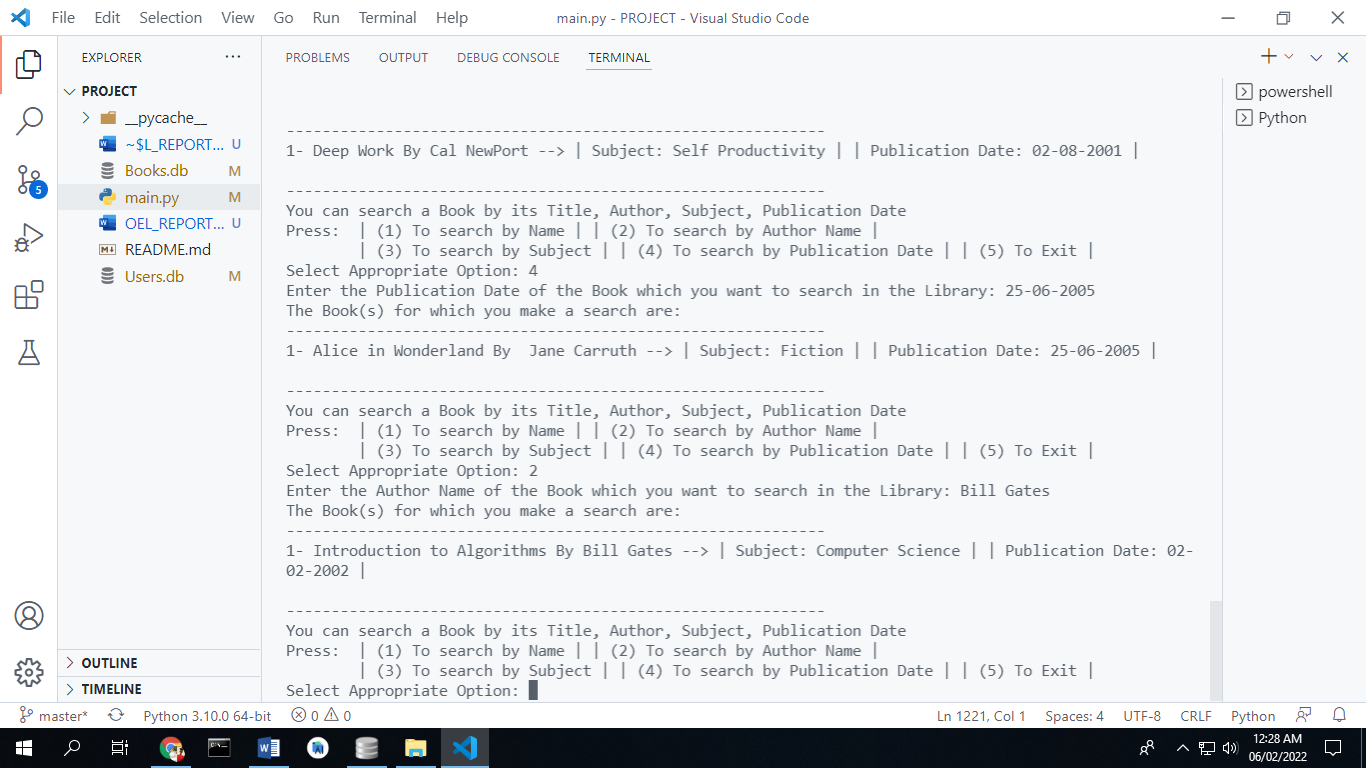
Below are attached the searching mechanisms screenshots for successful and unsuccessful searches:

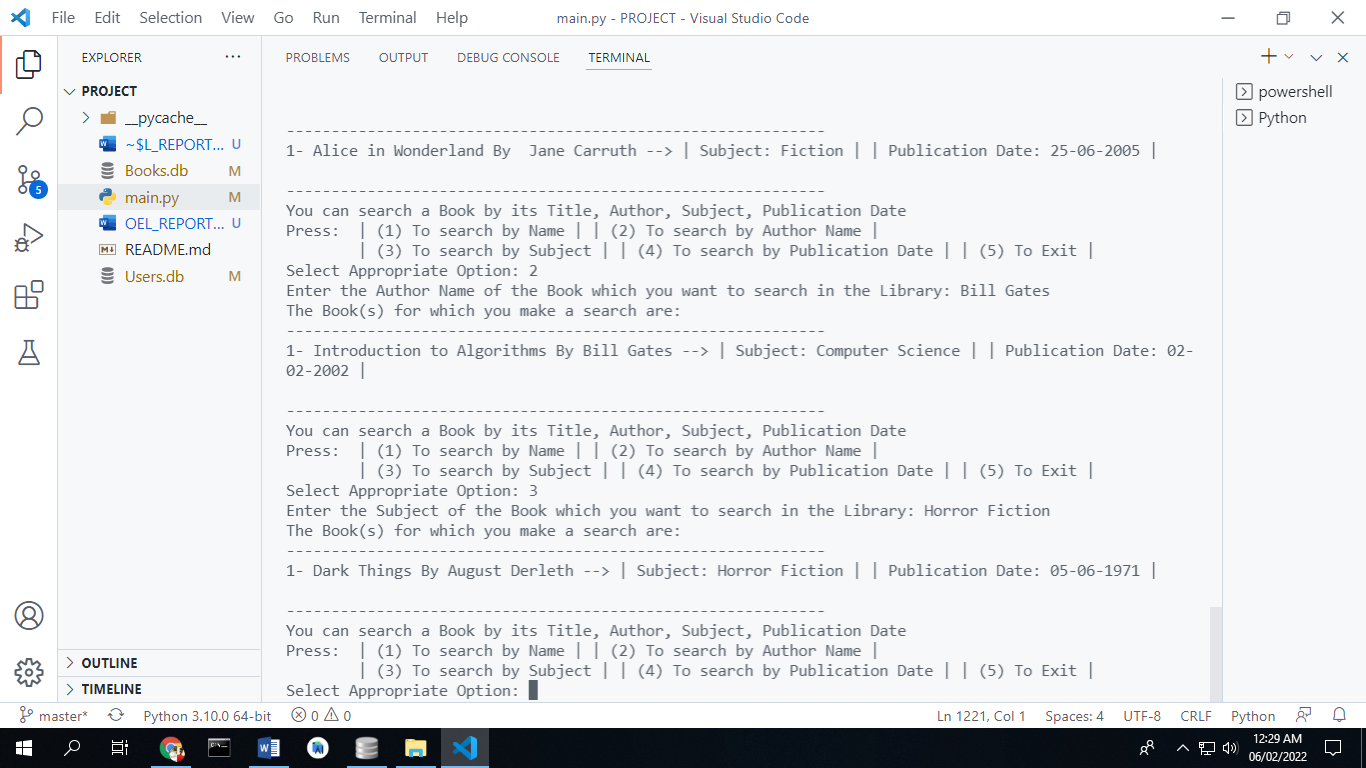
**. Successful Searches:**

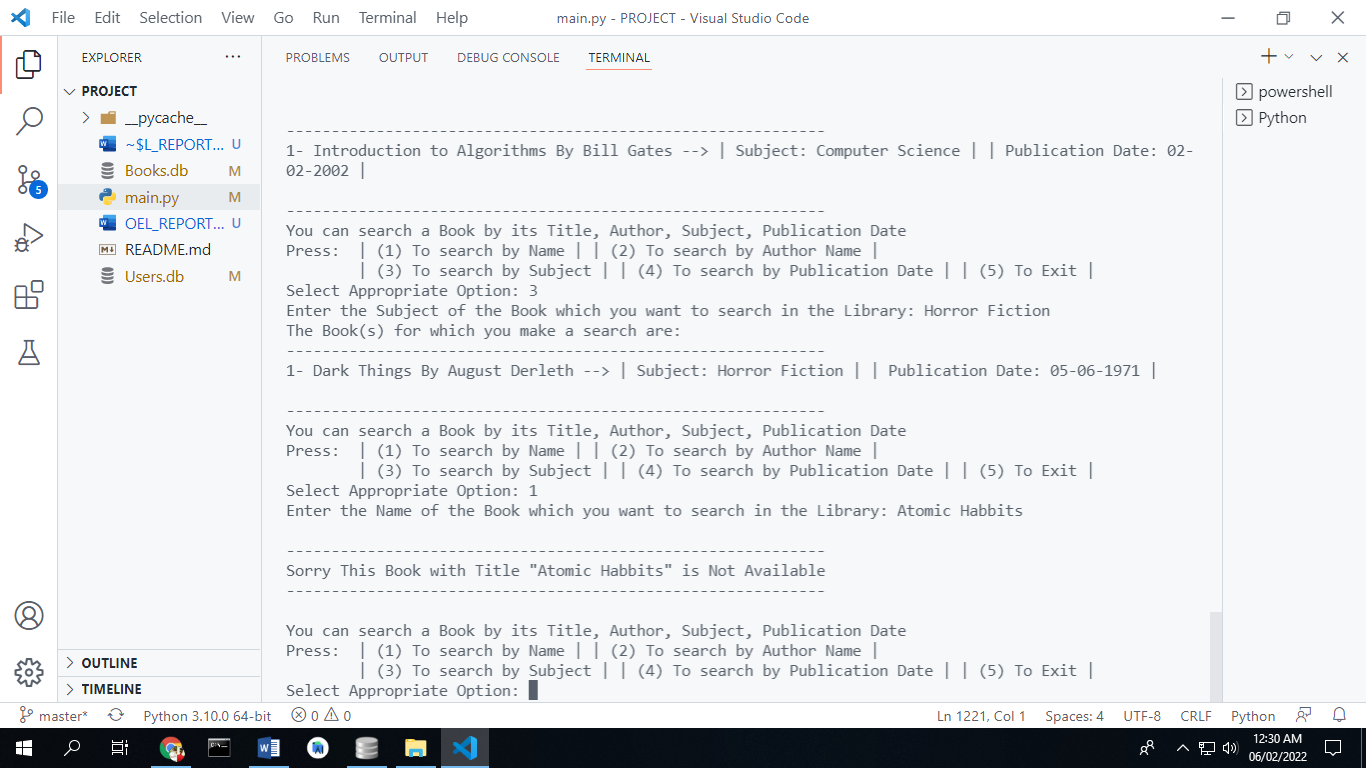


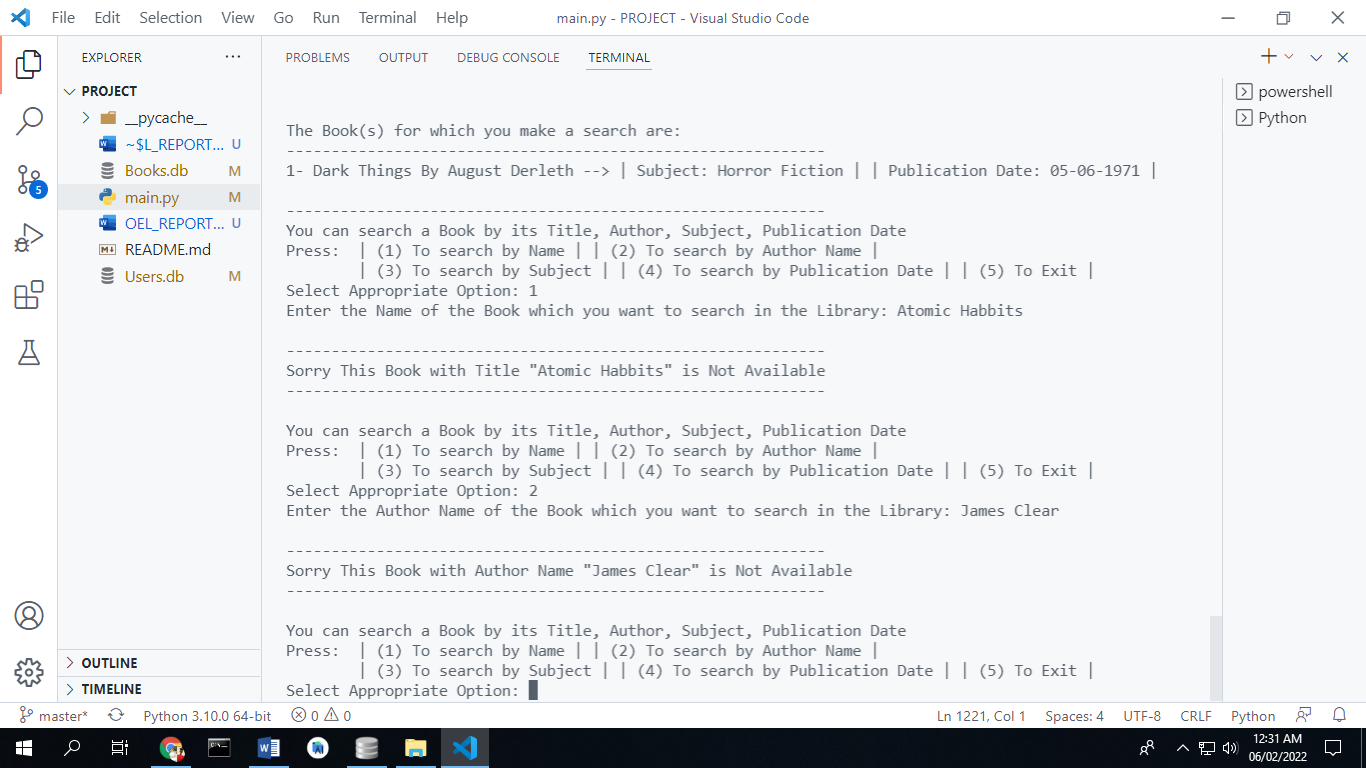


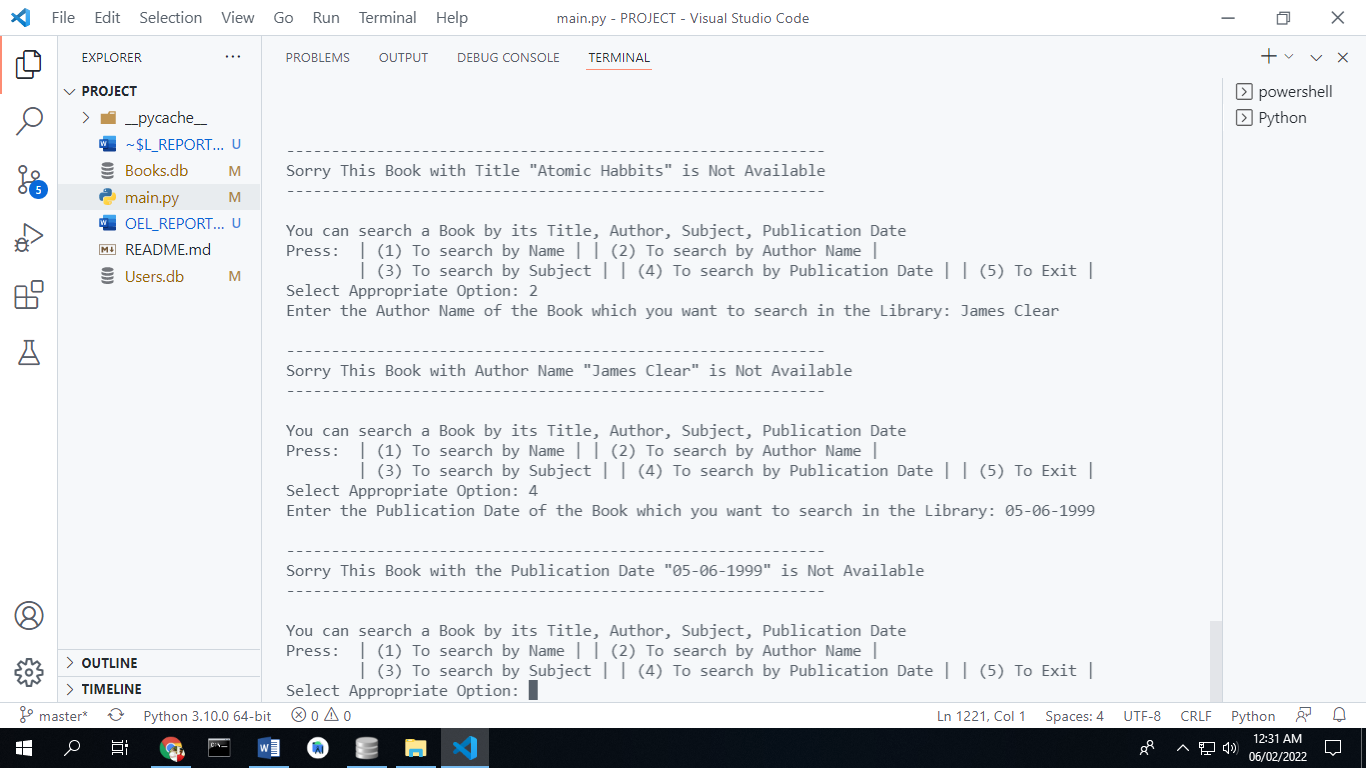


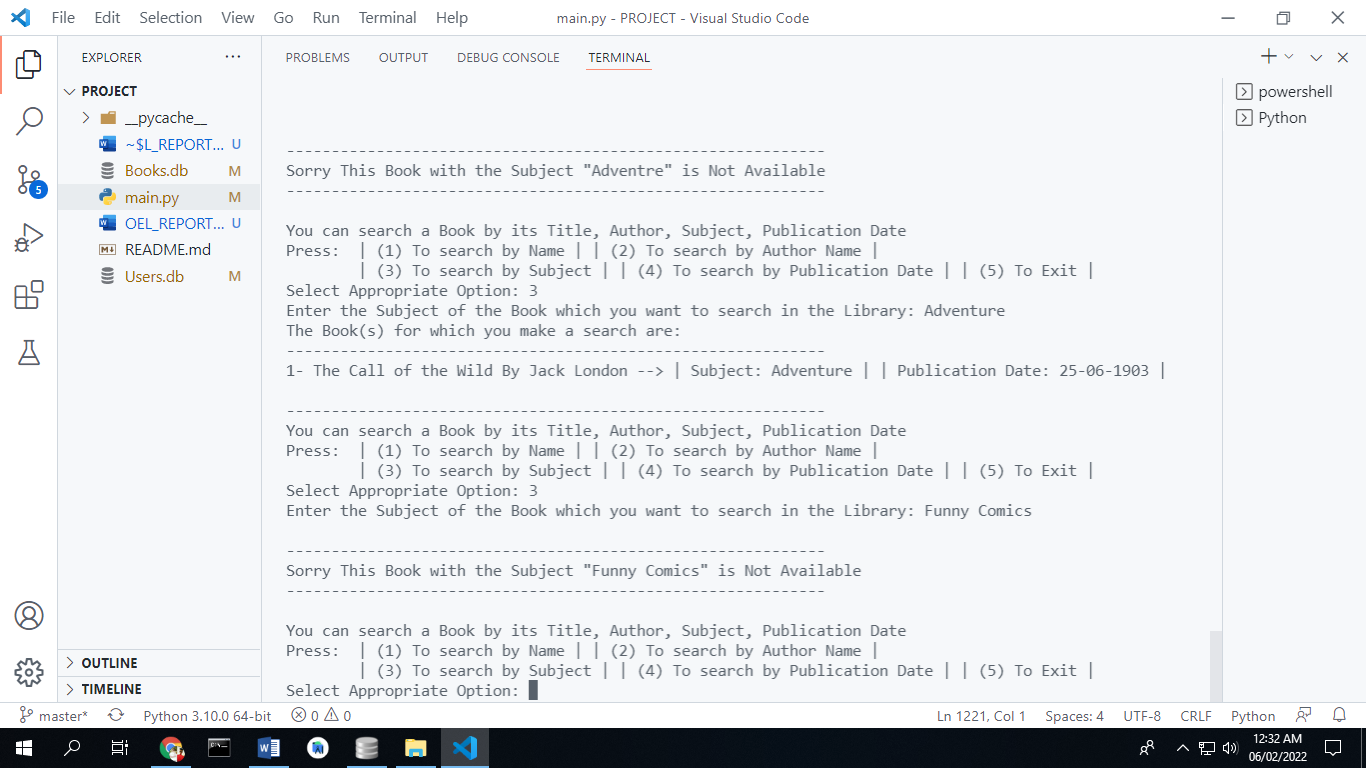


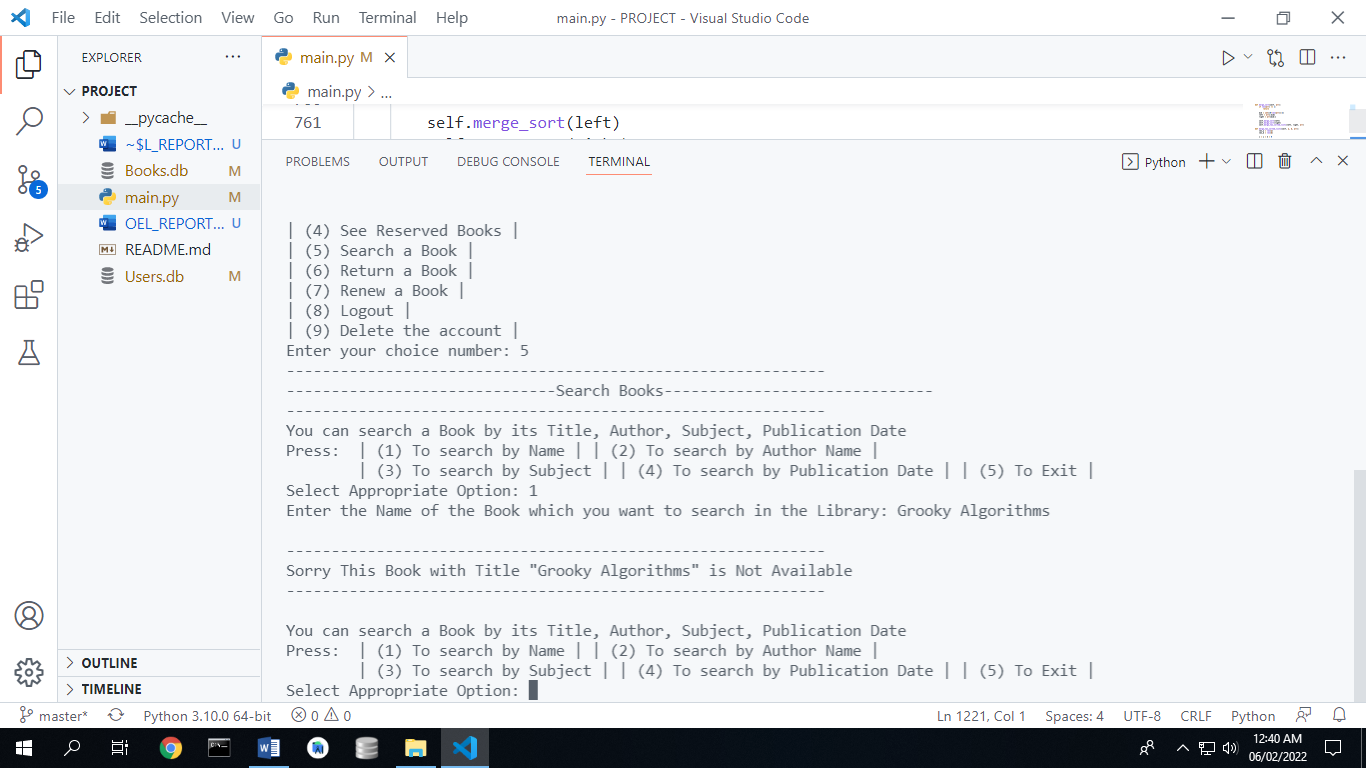
**. Unsuccessful Searches:**











* **A Guide to use the Application:-**

In order to use the application just download the **main.py, Users.db, and Books.db** all three files in one directory and run the **main.py** file. To login as an admin enter the credentials given below:

1. Email: [raffayadmin@gmail.com](mailto:raffayadmin@gmail.com)
2. Password: RaffayAdmin@#123

To login as a user you can just signup by creating a new account or can enter these credentials of an already user made account:

1. Email: [alikhan@gmail.com](mailto:alikhan@gmail.com)
2. Password: AliKhan@#123