Abstract:

To operate normally, every system would require management. In this current era of technology, a Library accomplishes the same thing for its smooth work flow. This program was designed to keep the data safe and accessible. Consider how many books from the library could be lost as a result of a data breach, as well as the data of the user, whether he is a library member or not. With only one click, the Admin can quickly track the user's record as well as the books that have been issued or are available in the library.
As a result, by using what we learned in class about Python OOP and Data Structure to develop a program that solves such issues in a library.

Introduction:

The project is intended to maintain track of two key bodies: a library's administrator and a library's user. This project keeps track of all of the books in the library, whether they are issued or returned, as well as all of the library's members. If a new book is added to the library, the administrator can insert it, or delete it if it is no longer available. The user can ask the admin to issue them a book, check if the book has been issued to them, and return the book they have received from the admin. Each user will be assigned a unique user ID.

Objectives:

1. The main object of the library management system is to manage details of members, issues, books, student.
2. Library Management system helps in maintaining data of books issued to learners.
3. Library Management system helps in maintaining data of books available in the library.
4. This helps librarians spot any particular book at any given time in the library.
5. It helps in keeping track of the books, catalogues, magazines, etc.
6. This system increases the efficiency of the librarian and better management of the library.
7. This system completely automates all your library’s activities.
8. Easy search of the desired book from the library.

Implementation:

- The program can be implemented by creating an algorithm that allows the user to add, remove, view, assign, and return a book to a library member using the Doubly link list, with some features like Stack, Queue, and File Handling.
- Furthermore, we will develop a login mechanism to determine whether or not a user is a library member, which will be done using the file handling system.

Methodology/ Plan development to solve the Problem:

Because we need to insert, delete, and search the Book, we're implementing the Double Link List, which is more efficient than alternative data structures. Because we have a constant time complexity of big O(1) for such functions in linked lists, it is easier to perform tasks like insertion and deletion. To create our project, we'll utilise the following function.

- **Login**

  At login menu we will be asked to choose if we are an Admin or we are a user. If we choose admin the we will be taken to admin login check to verify if we are admin or not and it will be checked by entering the user name and the password and the same is true for user. If we are a use then we will be asked to enter our login info so we can perform the user specific functions.

- **Admin Login Check**

  If we choose that we are admin the we will be asked to Enter the user name and the password if it the entered user name and password is true then we will be asked to choose a function we want to perform e.g Add Book.

  - If Admin:

    ⇒ **Add Book**
    Using the function Insertnewbook we add node as a book in the DLL and at the same time that node is added in a file name Book.py to keep record of the books.

    ⇒ **Del Book**
    Using the function Deletebook we delete the node in the DLL and also from the file named Book.py

    ⇒ **Issue Book**
    In Issue Book Function First we will check the request of the user who want to issue the book the first person that requested will be issued the book first and after that every other in a queue will be severed. After taking the request we will issued the book to the user and del the book from the files and add it in a dictionary to keep record of book issued.

    ⇒ **View Books**
    In View Book Function we will traverse the whole DLL and check each node (node are books) if the book user looking for is a node of DLL then we will return that the book is available in the Library.

    ⇒ **View Record of Issued Books**
    View Record of the Issued function will give us a list of the book that has been issued to the users and currently not available in the library.

- **User Login Check**

  This Function will check the user if the user is the member of the library by check it user id , user name and password if it is true then the user can have access to the library and we will able to perform the specific functions e.g Request for issue Book.

  - If User:

    ⇒ **Request For Issue Book**
    In this function the user will be able to request for a book and will wait for his/her turn. If the user is the first in the queue then he/she we will issued the book first.

    ⇒ **Return Book**
    The return book function is used for returning the book by the user. The user can easily return the book to the library.

    - ◊ **Reissue:**
      The Book user recently return if he/she wants to reissue it the user can easily use this function.

⇒ **View Book**

In View Book Function we will traverse the whole DLL and check each node (node are books) if the book user looking for is a node of DLL then we will return that the book is available in the Library.

⇒ **View if Book is Issued or not**

This function is used by the user to check is the admin issued him/her the book or not.

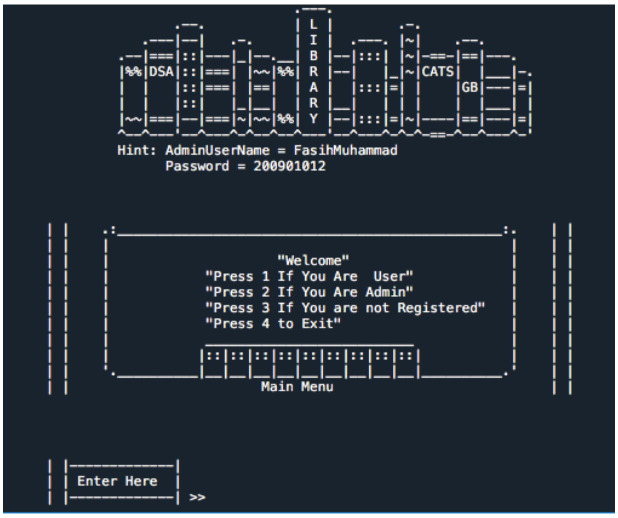
- **Register Your-Self**

  If the User is not register they cannot login in to issue a book so before login we have a function for those user who are not register but want to be a member of the library then they can register their self easily.

Output:

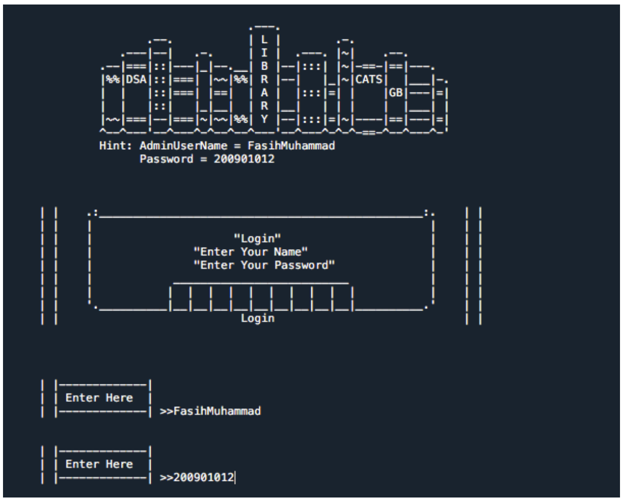
The Output is from the python IDE Spyder we build our project on this IDE and run it as the given ouput shown and we have shown a little interface and functionality of out program.

The Main Menu



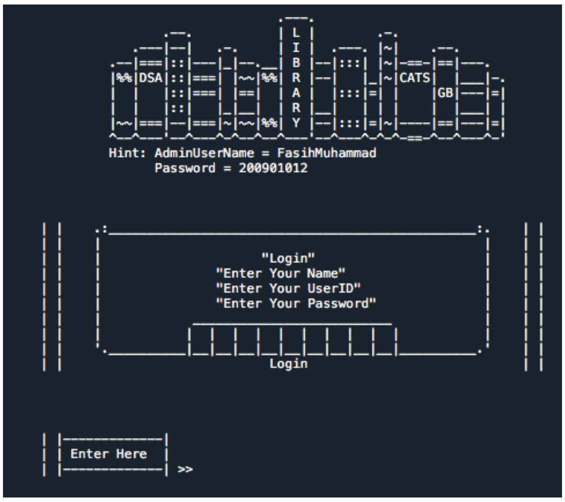
```
Hint: AdminUserName = FasihMuhammad
      Password = 200901012

                    "Welcome"
          "Press 1 If You Are  User"
          "Press 2 If You Are Admin"
          "Press 3 If You are not Registered"
          "Press 4 to Exit"

                   Main Menu

 Enter Here
            >>
```

If we choose 2 from Main Menu



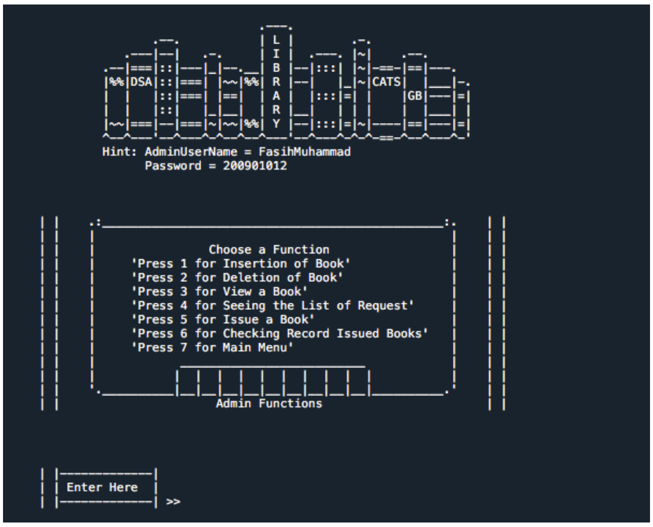
```
Hint: AdminUserName = FasihMuhammad
      Password = 200901012

                    "Login"
               "Enter Your Name"
               "Enter Your Password"

                     Login

 Enter Here
            >>FasihMuhammad

 Enter Here
            >>200901012
```

If we choose 1 from main menu



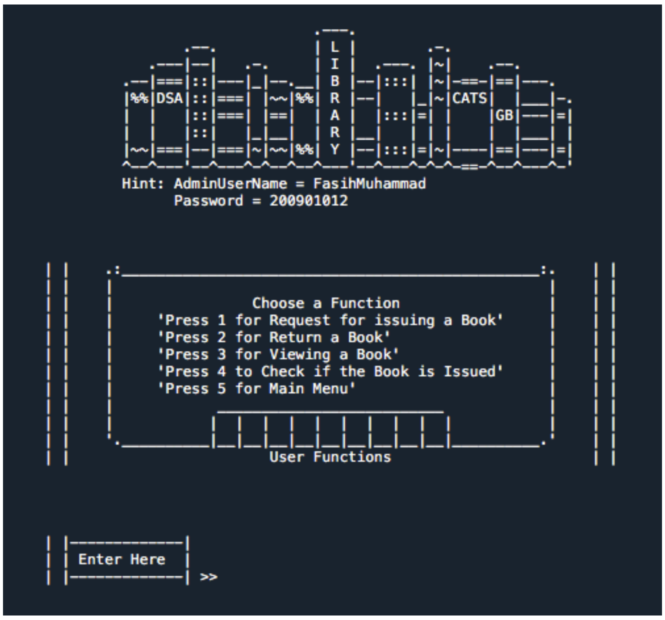
```
Hint: AdminUserName = FasihMuhammad
      Password = 200901012

                    "Login"
               "Enter Your Name"
               "Enter Your UserID"
               "Enter Your Password"

                     Login

 Enter Here
            >>
```

Function we can perform as an Admin



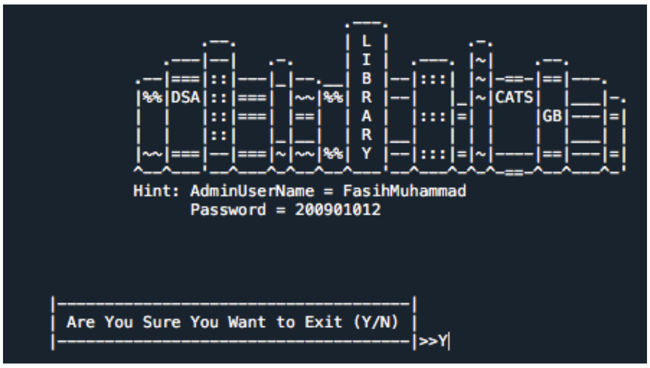
```
Hint: AdminUserName = FasihMuhammad
      Password = 200901012

        Choose a Function
   'Press 1 for Insertion of Book'
   'Press 2 for Deletion of Book'
   'Press 3 for View a Book'
   'Press 4 for Seeing the List of Request'
   'Press 5 for Issue a Book'
   'Press 6 for Checking Record Issued Books'
   'Press 7 for Main Menu'
          Admin Functions

| |------------|
| | Enter Here |
| |------------| >>
```

Function you can perform as a User

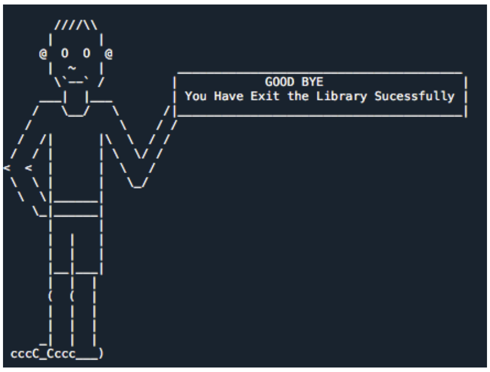


```
Hint: AdminUserName = FasihMuhammad
      Password = 200901012

        Choose a Function
   'Press 1 for Request for issuing a Book'
   'Press 2 for Return a Book'
   'Press 3 for Viewing a Book'
   'Press 4 to Check if the Book is Issued'
   'Press 5 for Main Menu'
          User Functions

| |------------|
| | Enter Here |
| |------------| >>
```

Exit



```
Hint: AdminUserName = FasihMuhammad
      Password = 200901012

|-----------------------------------|
| Are You Sure You Want to Exit (Y/N) |
|-----------------------------------|>>Y
```



```
                 GOOD BYE
| You Have Exit the Library Sucessfully |

cccC_Cccc___)
```

Specifications:

1. For the insertion and deletion of books in the library, as well as traversal, a doubly link list is implemented.
2. File Handling is used to keep track of Books and User Information.
3. Each Module has a distinct set of functions that are interlinked according to the requirements.
4. Loops are used to keep the programme running until the user wishes to depart.
5. Design several Ascii logos and artworks to provide our code a user interface.
6. Queue and Stack are also used in DLL code.
7. Classes, break statements, if-else, functions, loops, nested loops, file handling, and modules were all presented in the code.

Conclusion:

This project simplifies library work, making it more convenient than the manual approach. The digitalization of the management process makes it more efficient, secure, and dependable. The software was fully tested by inserting dummy data, and it performed all functions as expected. For modest, ordinary libraries, the software is ideal. It can have more capabilities in the future, such as requesting several books, asking delivery of a certain book to the user's house, or adding that if the user does not return the book by a given date, he or she will be punished, and we can also add features like auto-issuing the book to the user.