LIBRARY MANAGEMENT SYSTEM

A C++ project using file handling and functions (CUI-based)

Imaya senuri

Content

01. Introduction

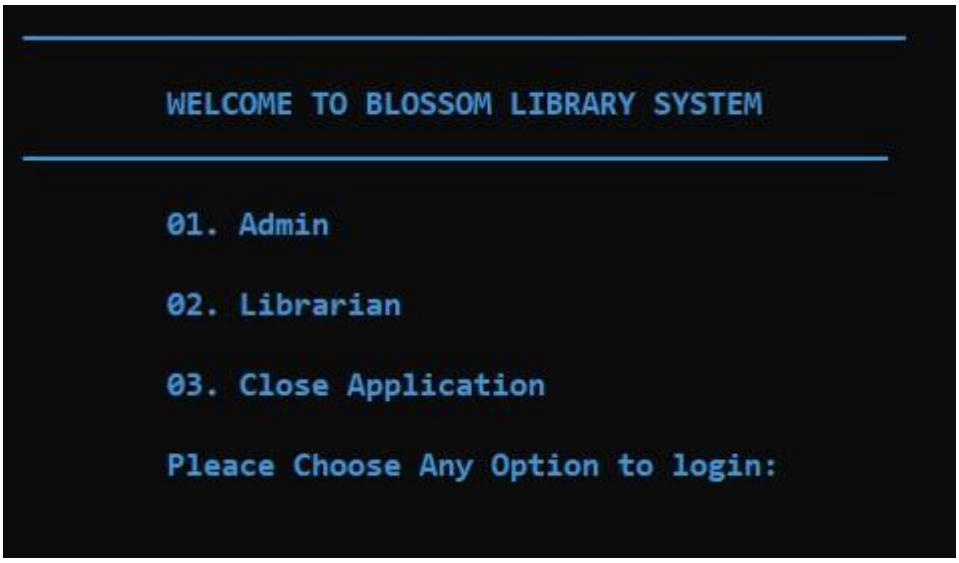
The blossom library system can be used by admins and librarians. This library system has requested that admin can sign up librarians, admin can change the fine rate and change the maximum borrowing limit.

Also, the librarians are able to log into the system with the username and password provided by the admin and to be able to add and delete books to the book list. Also able to search and update books if needed.

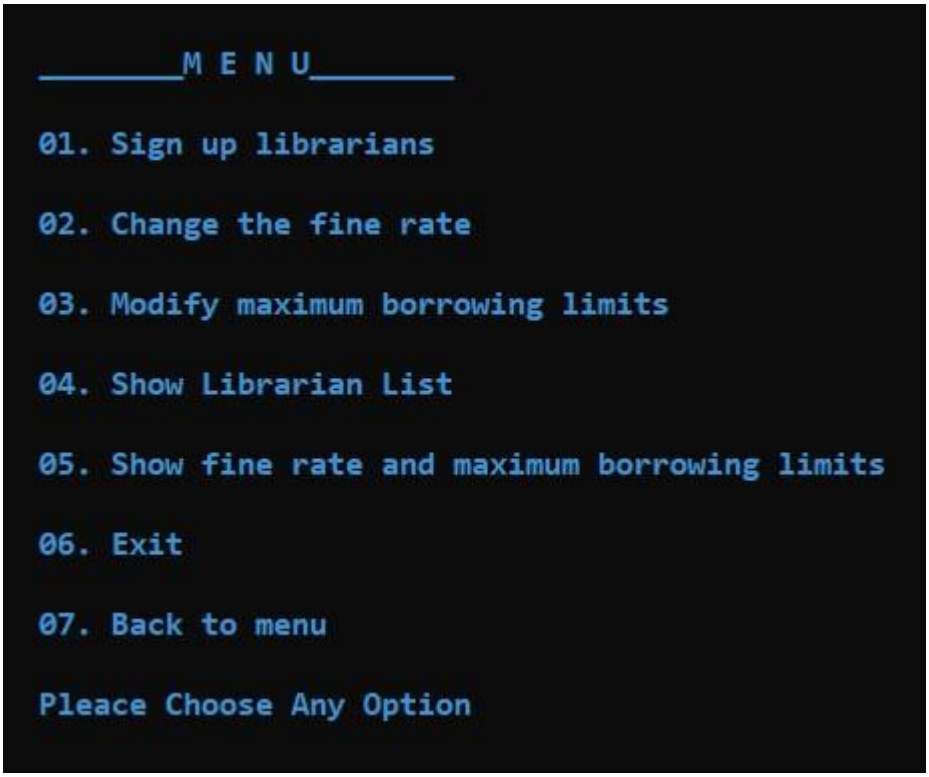
And also librarians can issue books and accept returned books. Librarians also maintain a Borrowers List where Borrowers can be add, delete, search and update

02. Evidence for the system implementation.

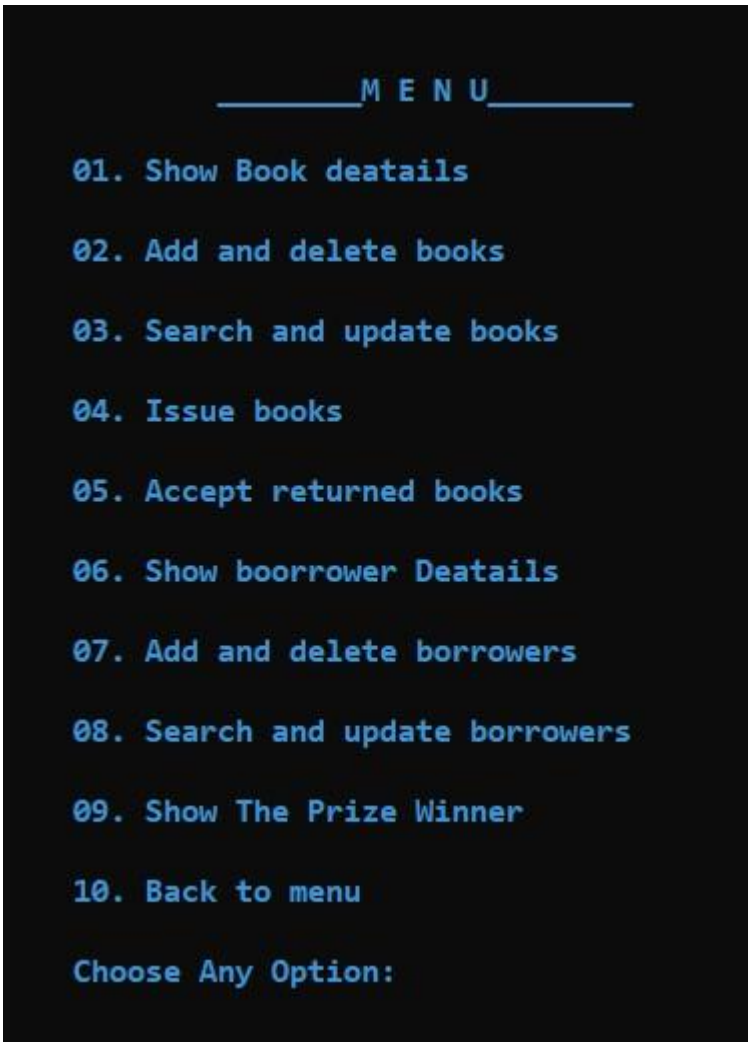
i) Main Menu



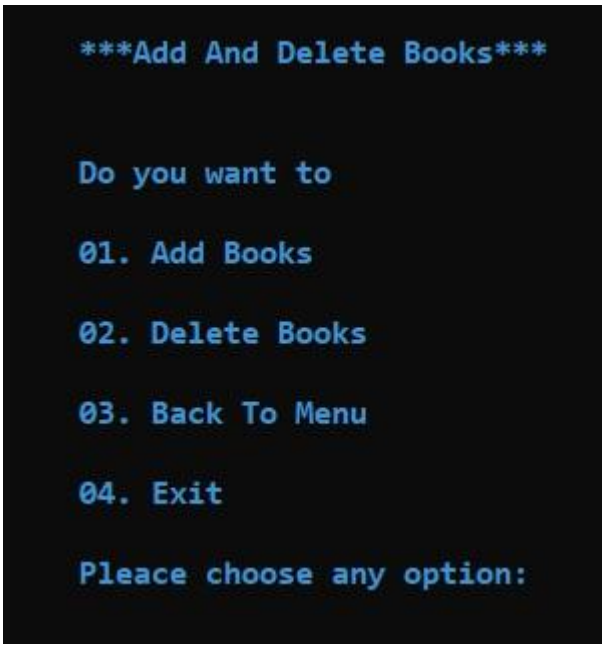
ii) Admin Task Menu



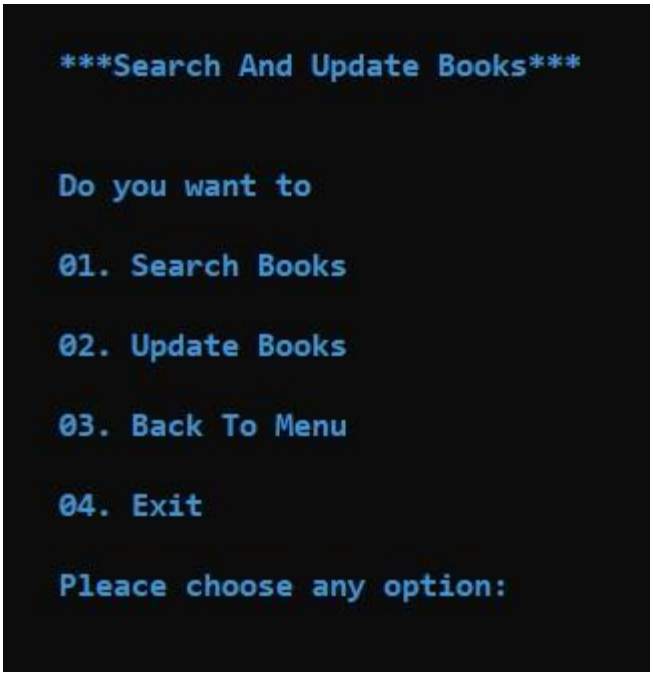
iii) Librarian Task Menu



iv) Librarian task add and delete books menu.

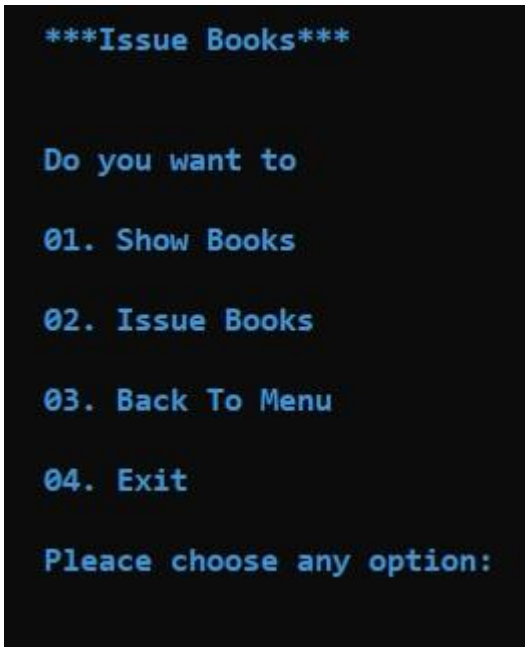


v) Librarian task search and update books menu.

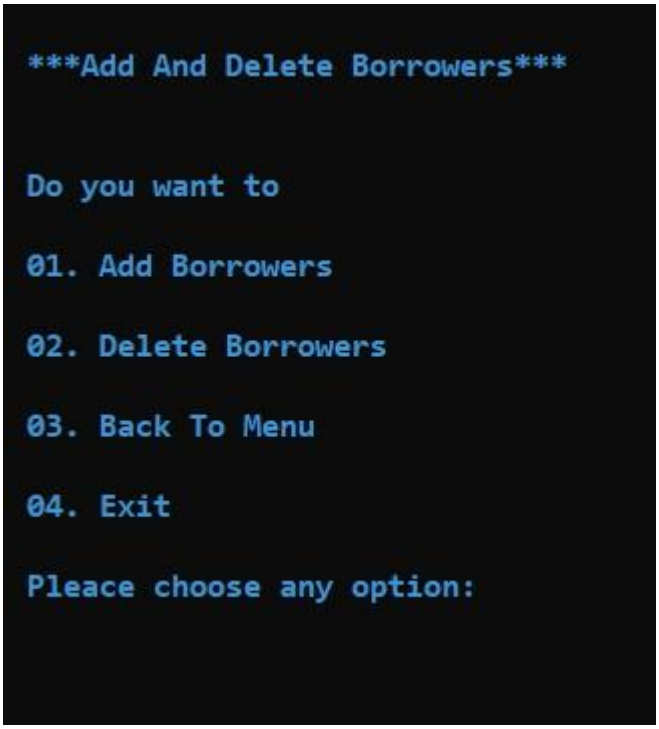


vi) Librarian task search and update books , update books menu

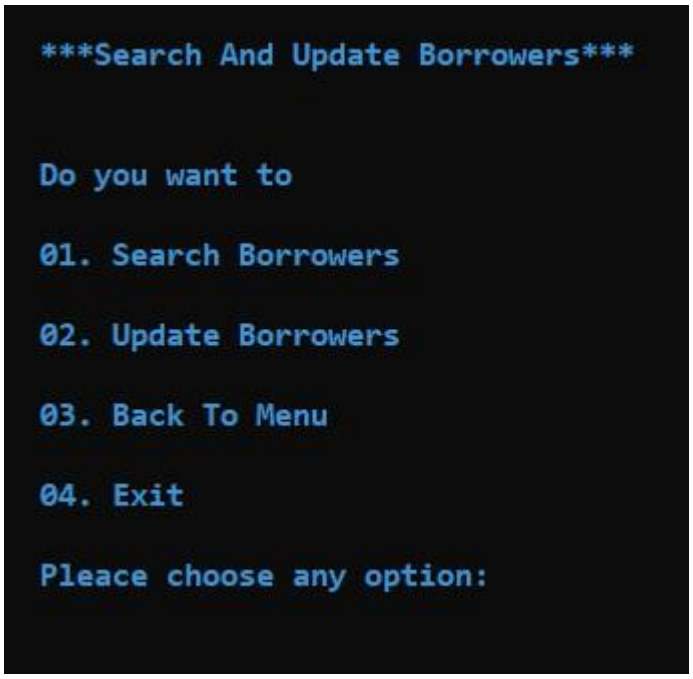
vii) Librarian task issue books menu.



viii) Librarian tasks add and delete borrowers menu.



ix) Librarian task search and update borrowers menu.



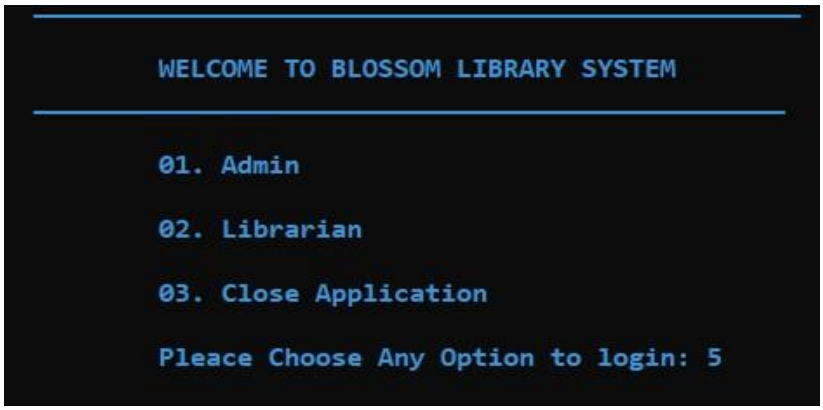
x) Librarian task search and update borrowers menu, update borrowers menu

03. Identify and implement opportunities for data validation and error handlings.

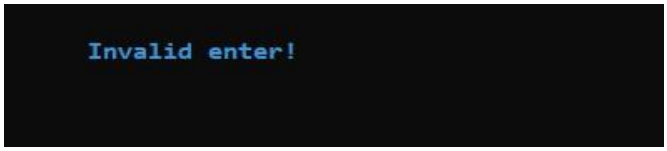
i) Displaying an error message when entering a value other than the given options 1, 2 or 3 in the main menu

```
default:{
    if(system("CLS")) system("clear");
    cout<<"\t\t\t\t\tInvalid enter!";
    menu1=1;
    break;
```

Command prompt code



Inserting incorrect value.



Displaying error message.

ii) An invalid password message will be displayed if a wrong password is entered when entering the admin password.

```
cout<<endl<<"\t\t\t\t\tEnter Admin password: ";
cin>>adminPassword;
cout<<endl;
if(system("CLS")) system("clear");
if(adminPassword==adminPasswordOriginal) {
else {
    cout<<"\t\t\t\t\tWrong password! Try again...";
    menu1=1;
}
```

iii) Displaying an error message when a wrong number is entered in the librarian task menu

```
default:{
    if(system("CLS")) system("clear");
    cout<<endl<<endl<<"\t\t\t\t\tInvalid enter!";
    menu1=1;
    break;
}
```

iv) Displaying an error message if the librarian enters an incorrect username and password during login

```
string line;
while(getline(signfile, line)) {
    istringstream iss(line);
    if(iss>>name>>nic>>fusername>>fpassword) {
        if(fpassword==password && fusername==username) {
            count=count+1;
        }
```

v) Displaying an error message if the librarian enter a wrong value for librarian task add and delete books menu.

```
else if(addOrDelete<1 || addOrDelete>4) {
    cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";
    addOrDelete=1;
}
```

vi) Displaying an error message if the librarian enter a wrong value for librarian task search and update menu.

```
else if(searchORupdate<1 || searchORupdate>4) {
   cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";
   searchORupdate=1;
}
```

vii) Displaying an error message if the librarian enter a wrong value for librarian task issue books menu.

```
else if(issuebooks<1 || issuebooks>4) {
   cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";
   searchORupdate=1;
}
```

viii) Displaying an error message if the librarian enter a wrong value for librarian task add and delete borrowers menu.

```
else if(addDeleteBorrowers<1 || addDeleteBorrowers>4) {
   cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";
   addDeleteBorrowers=1;
}
```

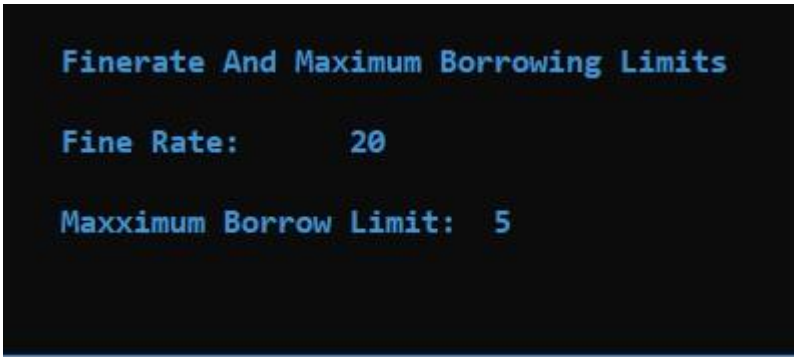
ix) Displaying and error message if the librarian enter a wrong value for librarian task search and update borrowers.

```
else if(searchORupdate<1 || searchORupdate>4) {
   cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";
   searchORupdate=1;
}
```

04. Additional features.

i) If the admins want , They can see the details of the librarians.

ii) If the admins want, They can see fine rate and Maximum borrow limits.



iii) Librarians can see, book details whenever they want.

,,,,,,,,,____BOOK LIST____,,,,,,,,,

ID Of The Book	Name Of The Book	Author Of The Book	Genre Of The Book	Availability status
	malagiya aththo	ediriweera sarathchandra	love	
001	kaliyugaya	martin wickramasinghe	fji	unavailable
002	kaliyugaya	martin wickramasinghe	fji	unavailable
003	madol duuwa	martin wickramasinghe	child	unavailable
004	heen saraya	kumarathunga munidasa	child	unavailable
005	baddegama	a.p. gunarathna	child	unavailable
006	kaliyugaya	martin wickramasinghe	chil	unavailable
007	gamperaliya	martin wickramasinghe	child	unavailable
008	yuganthaya	martin wickramasinghe	novel	unavailable
009	ape gama	martin wickramasinghe	novel	unavailable
010	malagiya aththo	ediriweera sarathchandra	love	unavailable
006	kaliyugaya	martin wickramasinghe	fji	unavailable

iv) Librarians can see , borrower details whenever they want.

---Borrower's Deatails---

Borrow Id	Borrower's Name	Joined Date	Contacts	Numbers Of Books Read
bb001	meesha	2023.01.31	0775555596	
bb002	nilu	2023.02.03	0775896879	1
bb003	amali	2023.05.20	0775634896	

v) At the end of the year, the library awards a prize to the person who reads the most books, and librarians can see who has qualified to win the prize at any time.

05. test plan to support the documentation of the testing process.

Project Case study	
Test Plan ID	001
Brief Introduction about the system.	This is a library system. This system can manage admins and librarians.
Test Objectives	1. Add data 2. Delete data 3. Output data 4. Search data 5. Update data
Features to be tested	1. Add librarian's data 2. Update fine rate 3. Update maximum borrowing limit 4. Show maximum borrowing limits and fine rate 5. Output book details 6. Add book details 7. Delete book details 8. Search book details 9. Update book details 10. Issue books 11. Accept returned books. 12. Output borrower details 13. Add borrower details 14. Delete borrower details 15. Search borrower details 16. Update borrower details 17. Output prize winner

Test Environment	Laptop , Dew C++ ,
Test Approach	Black box
Testing Tasks	Test planning, Test Design, Test development, Test execution, Test evaluation
Test deliverables	Test plan, Test Environment, Test summary, Test result, Test Evaluation report
Schedule	2023.08.17 01.40 a.m

06. Test cases for the system

Te ;t case 01	
Add librarian's data	Imaya Senuri
01	Black box
Add Librarian's first name and Librarian's last six digits of NIC to librarian data file	2023.08.17
Librarian details	1.47 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Log in as admin	01	Admin password-1234	Show admin menu	Show admin menu	pass
02	Add data of librarian	01	First name: mindi Last six digit of nic: 232323	Show signup successfully message and librarian's datas	Signup successful Message and librarian's details	pass

Test case 02	
Update fine rate	Imaya Senuri
02	Black box
Add a value for fine rate to fine rate file	2023.08.17
Fine rate	1.50 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	change fine rate	02	Fine rate value : 10	Show fine rate updated message	Fine rate updated	pass

Test case 03	
Update maximum borrowing limits	Imaya Senuri
03	Black box
Add a value for maximum borrowing limits for maximum borrowing limits file	2023.08.17
Maximum borrowing limit	1.55 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	change maximum borrowing limit	03	Maximum borrowing limit : 3	Show maximum borrowing limit updated message	Maximum borrowing limit updated	pass

Test case 04	
Show maximum borrowing limits and fine rate	Imaya Senuri
04	Black box
Display currently maximum borrowing limits and fine rate	2023.08.17
maximum borrowing limits and fine rate	2.02 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Show maximum borrowing limits and fine rate	04		Show currently maximum borrowing limits and fine rate	Show maximum borrowing limits and fine rate	pass

Te :t case 05	
Output book details	Imaya Senuri
05	Black box
Display all books’ records .	2023.08.17
Book details	2.04 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Show book details	05		Show book id, title, author, price, publisher, genre and availability status of all books	Show success book id ,title, author, price, publisher, genre and availability status of all books	pass

Test case 06	
Add book details	Imaya Senuri
06	Black box
Add book id, title, author, price, publisher, genre and availability status of the book to book details file	2023.08.17
Book details	2.05 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Add book details	06	Bookid : 006, book title : kaliyugaya ,author: martin wickramasinghe , price: 600 , publisher: sarasavi , genre : child , availability status :available	Show book details added message	Book details added	pass

Te;t case 07	
Delete book details	Imaya Senuri
07	Black box
Delete book details by book id	2023.08.17
Book details	2.08 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	delete book details	07	Book id: 013	Show book details have been deleted message	book details have been deleted	pass

Test case 08	
Search book details	Imaya Senuri
08	Black box
Search book details by book id	2023.08.17
Book details	2.11 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Search book details	08	Book id: 010	Show book details of book that entered id	Show book details	pass

Test case 09	
Update book details	Imaya Senuri
09	Black box
Update book details by book id	2023.08.17
Book details	2.14 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Update book details	09	Book id: 005	Show book details of book that entered id and book details file update	Show book details updated message and book details file updated	pass

Test case 10	
Issue books	Imaya Senuri
10	Black box
Issue books by book ID	2023.08.17
Issue details	2.40 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Issue books by book id	10	Book id: 007 Borrow id: bb002 Issue date: 2023.02.05 Due date: 2023.02.10	Show issue file updated message and book list updated message. Book details file update ,issue file update and borrowers; details file update	Issue file updated and booklist updated message . Book detail file updated, issue details file updated and borrower details file updated	pass

Test case 11	
Accept returned books	Imaya Senuri
11	Black box
Accept returned books by ID	2023.08.17
Issue details	10.40 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Accept returned books	11	Book id Returned day Borrow id Number of late days	Show fine rate for a day and fine rate	fine rate for a day and fine rate	pass

Test case 12	
Output borrower details	Imaya Senuri
12	Black box
Output currently borrow details from borrow details file	2023.08.17
borrower details	10.52 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Output borrower details	12		Show borrower details	Show borrower details	pass

Test case 13	
Add borrower details	Imaya Senuri
13	Black box
Add borrower details to the borrower data file	2023.08.17
borrower details	11.00 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Add borrower details	13	Borrow id: bb004 Borrow name: Gunathilaka Joined date: 2023.05.30 Contact number:0771212122	Show borrower details added message.	Borrower details added	pass

Test case 14	
Delete borrower details	Imaya Senuri
14	Black box
Delete borrower details by borrower id from borrower data file	2023.08.17
borrower details	11.06 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Delete borrower details	14	Borrow id: bb003	Show borrower details have been deleted message.	Borrower details have been deleted	pass

Test case 15	
Search borrower details	Imaya Senuri
15	Black box
Search borrower details by borrower id from borrower data file	2023.08.17
borrower details	11.12 a.m

Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Search borrower details	15	Borrow id: bb002	Show b002 borrower ‘s details	Show b002 borrower ‘s details	pass

Test case 16	
Update borrower details	Imaya Senuri
16	Black box
Update borrower details by borrower id from borrower data file	2023.08.17
borrower details	11.15 a.m

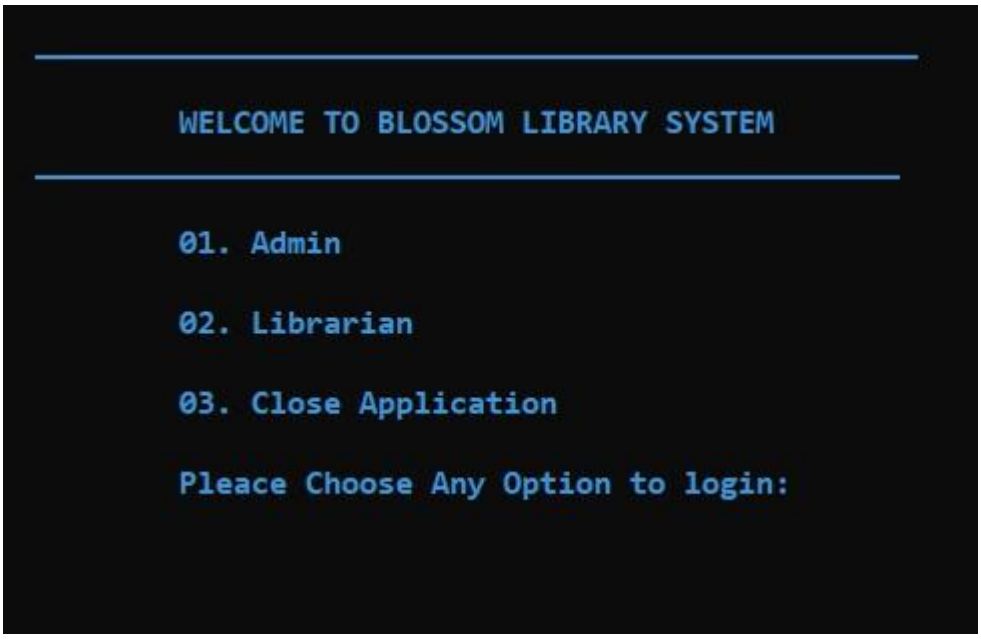
Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Update borrower details	16	Borrow id: bb00 Borrower's contact: 0778889990	Show updated successfully message	The borrower details have been updated	pass

Test case 17	
Output prize winner	Imaya Senuri
17	Black box
Showing the borrower who has read the most books	2023.08.17
borrower details	11.21 a.m

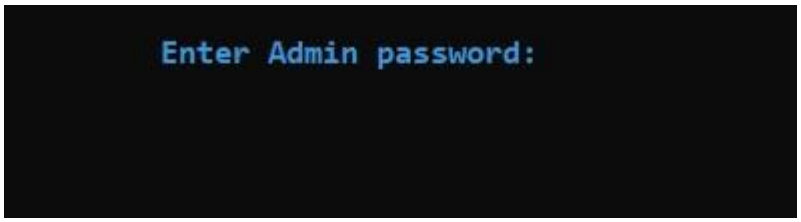
Step no	Test step	Test case ID	Test input	Expected result	Actual result	Test result
01	Show the prize winner	17		Showing the details of borrower who has read the most books	the details of borrower who has read the most books	pass

07. User documentation for the developed system.

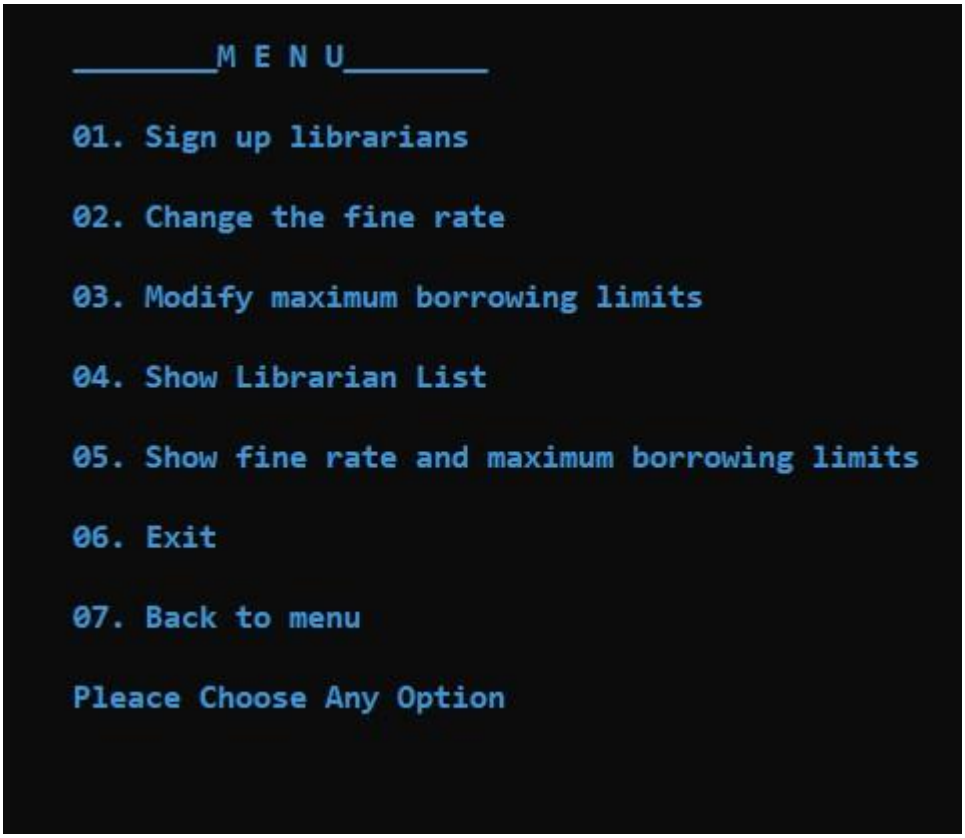
I. First you will be shown the following main menu. You can choose admin , librarian or close programe.



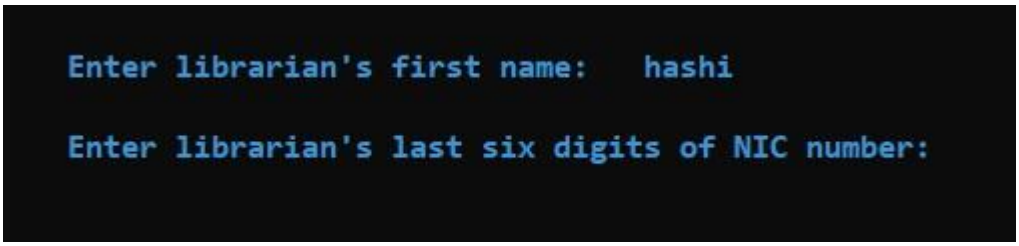
II. Then if you choose admi , you should enter admin password.



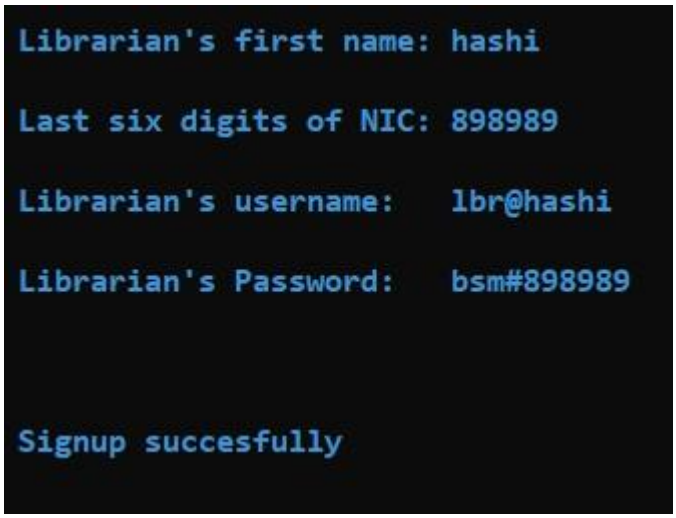
III. Then you can see admin menu. Here you have ther are seven options. You can choose anyone from there



IV. If you choose sign up librarian's , you should enter librarian's first name and the last six digits of librarian's NIC .

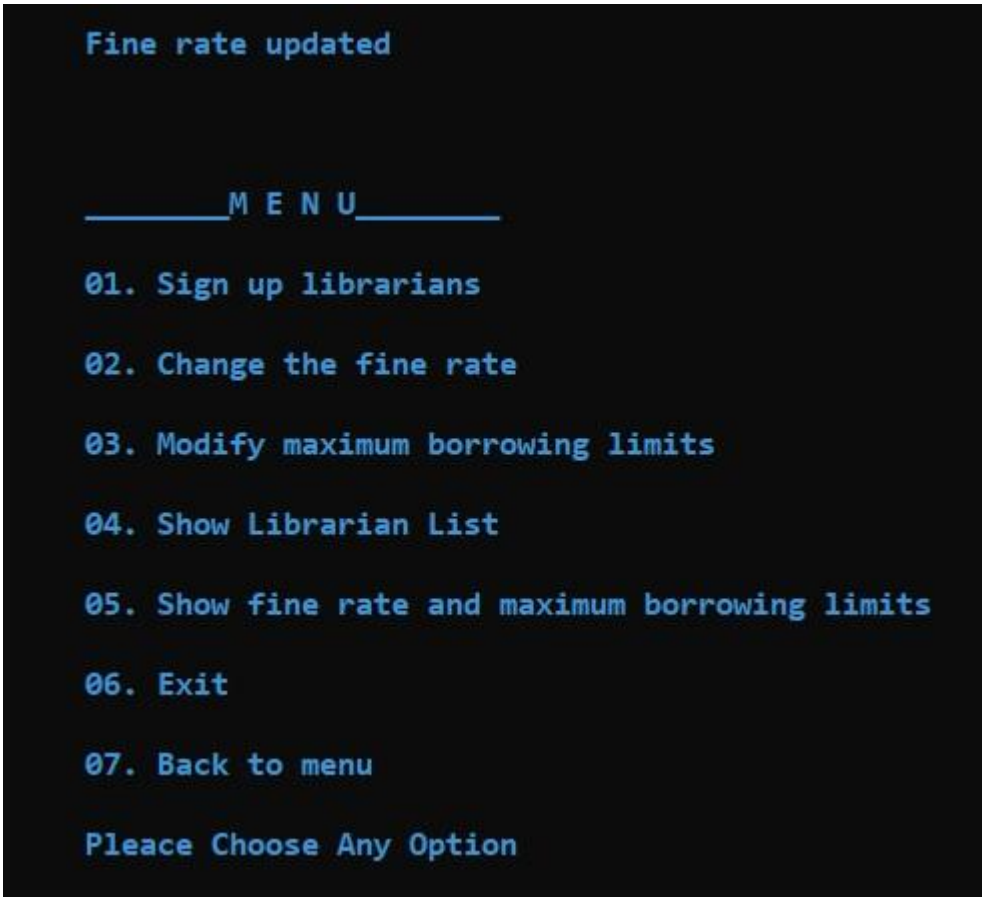


V. After entering the librarian's first name and last six digits of NIC , librarian's name, username and password will be displayed. And also displayed signup successfully message.

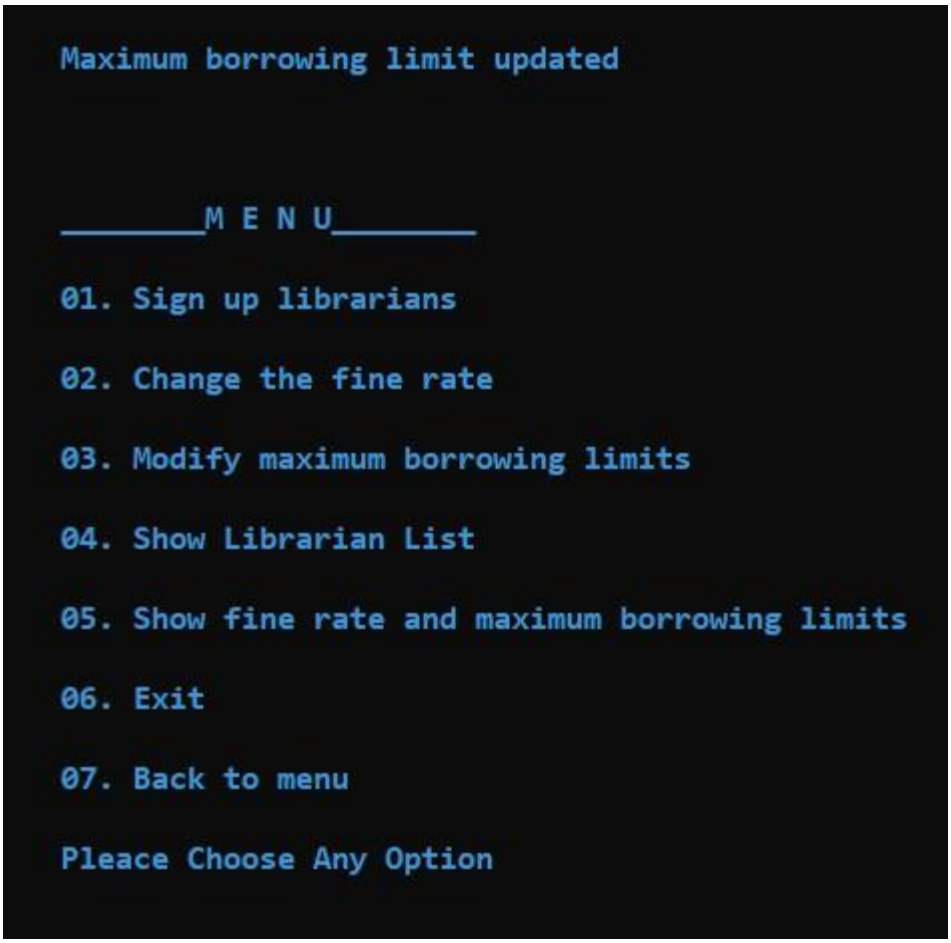


VI. After signup you will come to admin task menu automatically

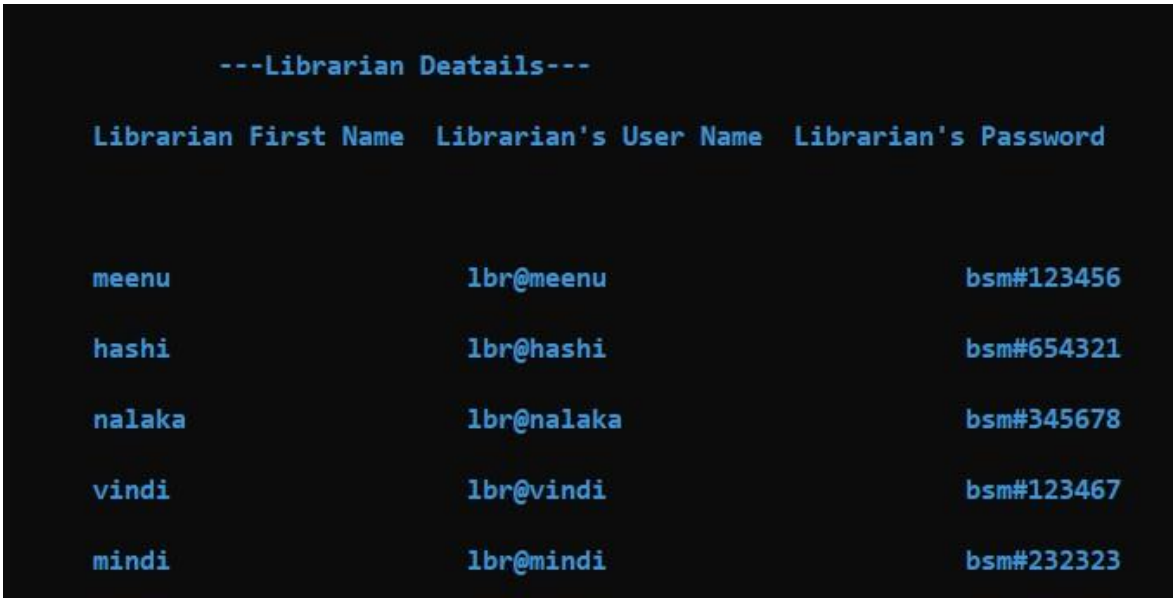
VII. Then if you choose change the fine rate, you should enter a value for fine rate. After enter a value , display a message, fine rate updated and again automatically show admin task menu.



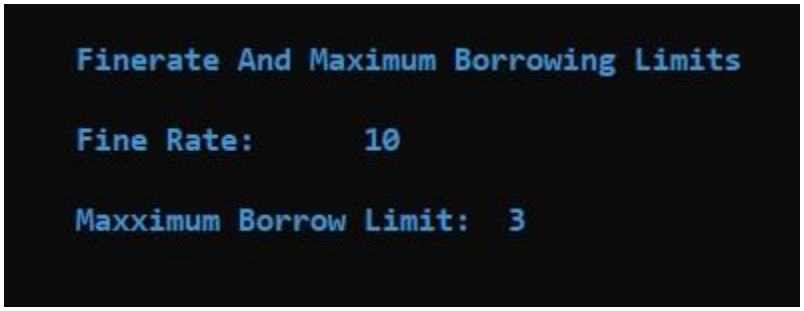
VIII. Then you choose modify maximum borrowing limits, you should enter a value for maximum borrowing limit. And after display a message maximum borrowing limit updated. And again display admin task menu.



IX. Then you choose show librarian list, display all librarian details And display librarian task menu



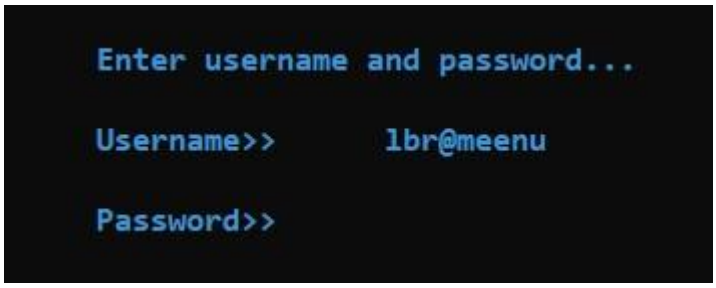
X. Then you choose show fine rate and maximum borrowing limits, will display currently fine rate and maximum borrowing limits . and also again display librarian task menu



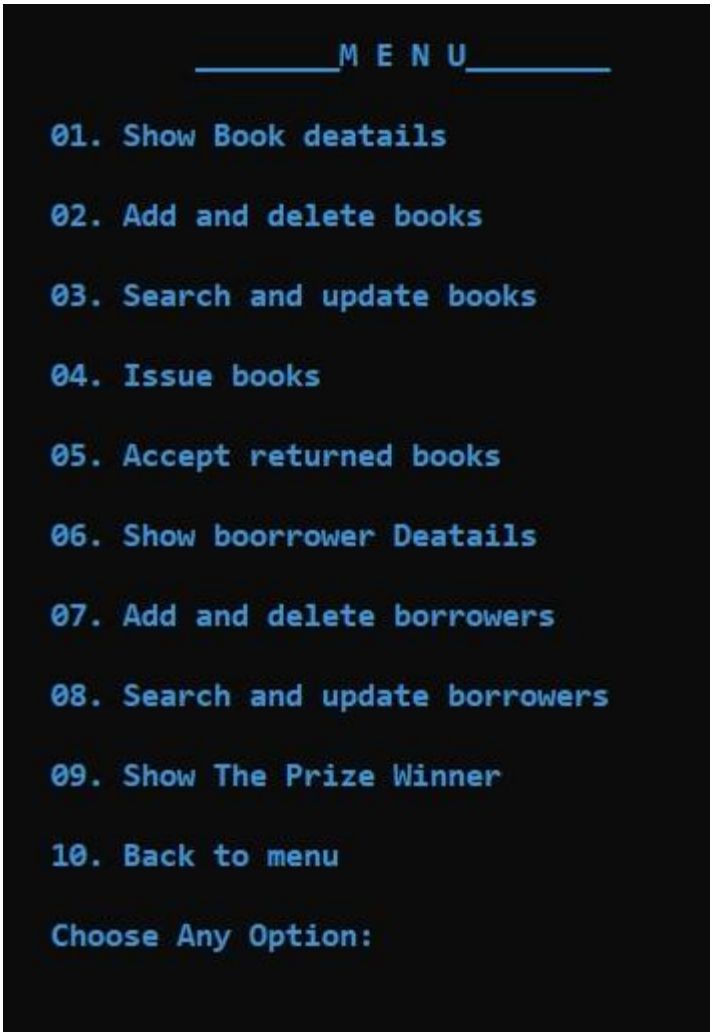
XI. Then you can exit or back to menu if you want

XII. If you choose back to menu , you will display main menu again and you can choose librarian

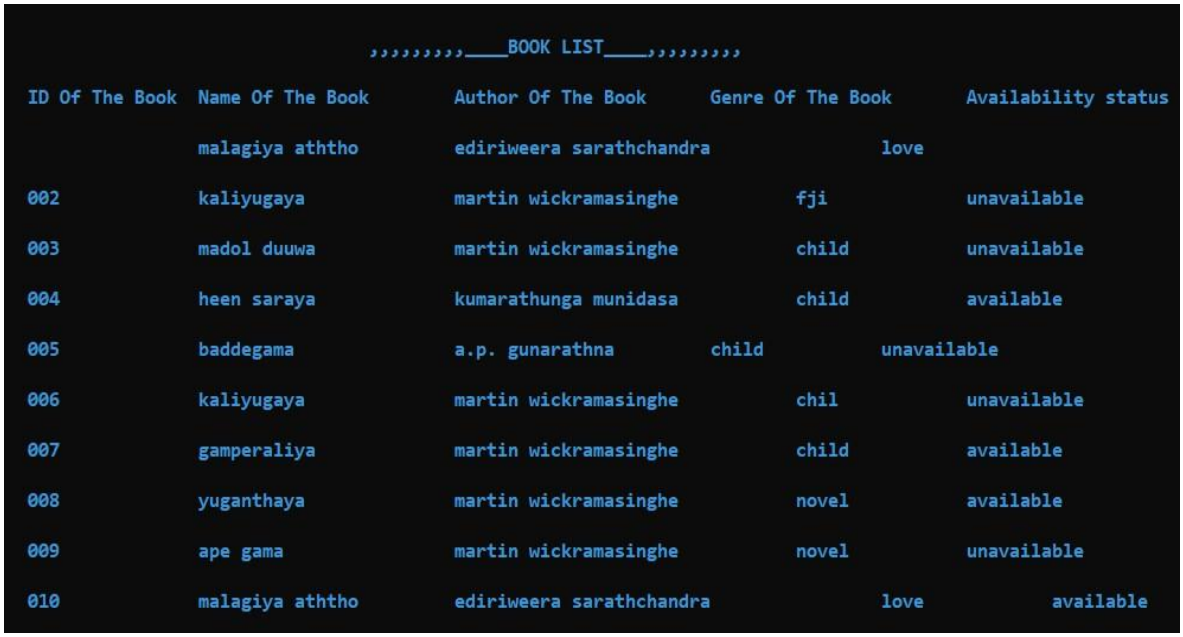
XIII. If you choose librarian you should enter librarian username and password correctly



XIV. After you enter username and password, you can see librarian task menu



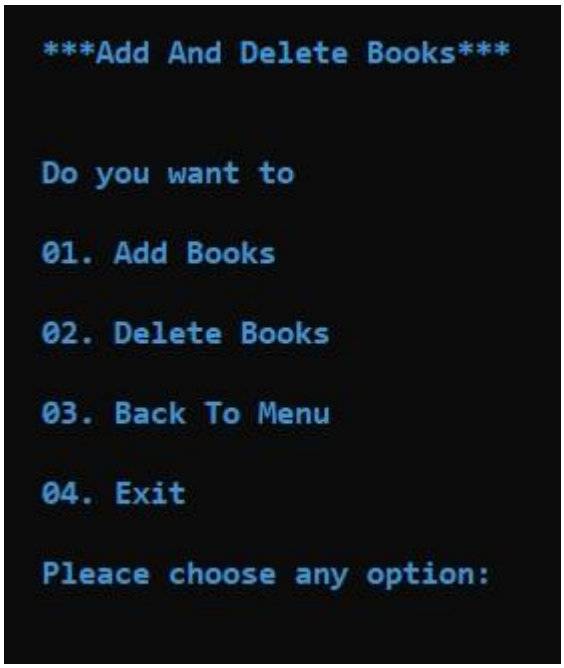
XV. If you choose show books, you can see details of all book records and librarian task menu again



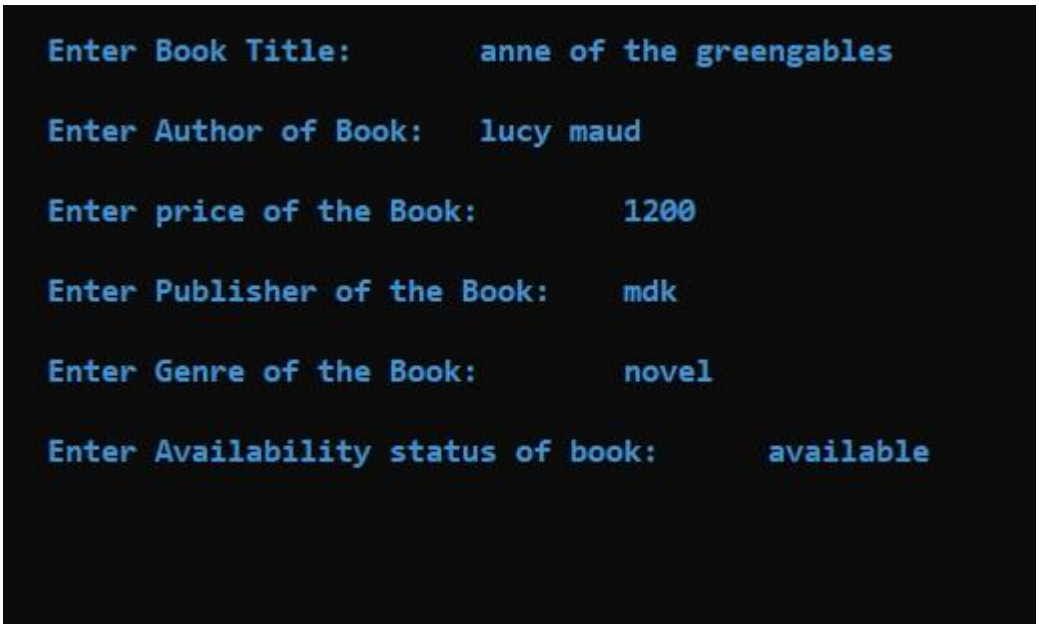
,,,,,,,,,____BOOK LIST____,,,,,,,,,

ID Of The Book	Name Of The Book	Author Of The Book	Genre Of The Book	Availability status
	malagiya aththo	ediriweera sarathchandra	love	
002	kaliyugaya	martin wickramasinghe	fji	unavailable
003	madol duuwa	martin wickramasinghe	child	unavailable
004	heen saraya	kumarathunga munidasa	child	available
005	baddegama	a.p. gunarathna	child	unavailable
006	kaliyugaya	martin wickramasinghe	chil	unavailable
007	gamperaliya	martin wickramasinghe	child	available
008	yuganthaya	martin wickramasinghe	novel	available
009	ape gama	martin wickramasinghe	novel	unavailable
010	malagiya aththo	ediriweera sarathchandra	love	available

XVI. If you choose add and delete books, you will go to add delete books menu.

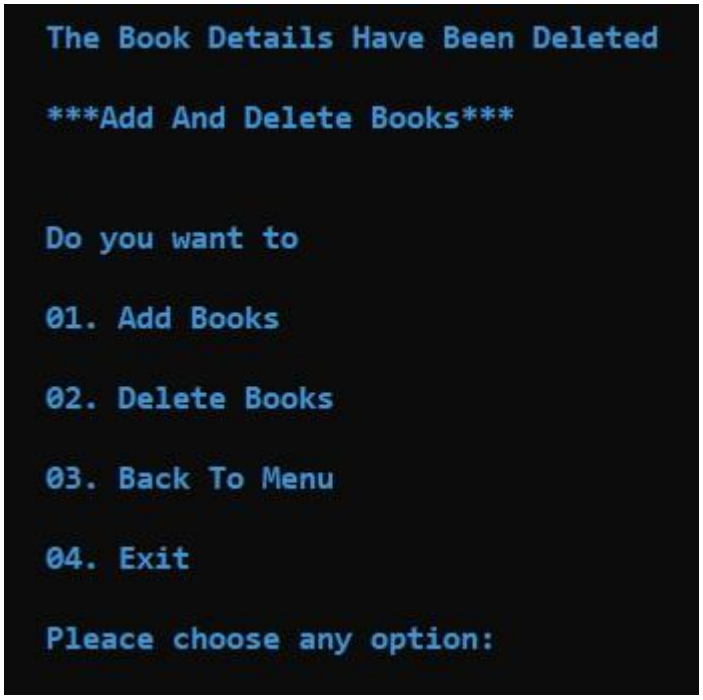


XVII. Then you choose add books, you should enter a id for the book, book title, genre , author , price , publisher and availability status.



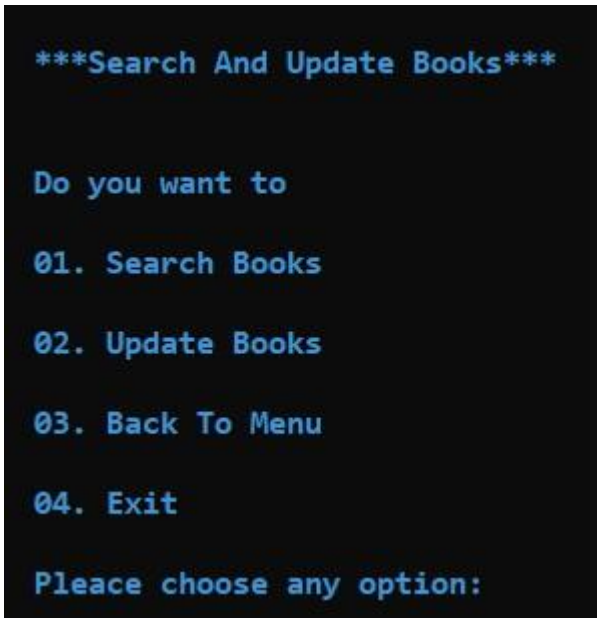
XVIII. Then display book details added message and add delete bokks menu

XIX. Then if you choose delete books, you should enter book id, and then display book details deleted message and add delete book menu.



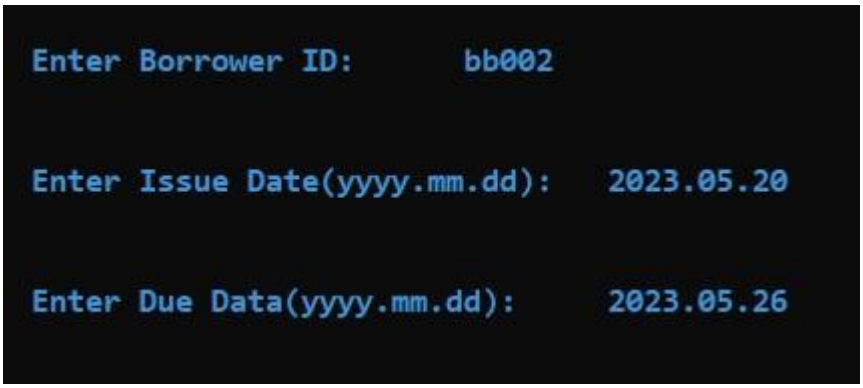
XX. Then you can exit or back to menu.

XXI. If you back to menu and choose search update books, you will display search and update books menu



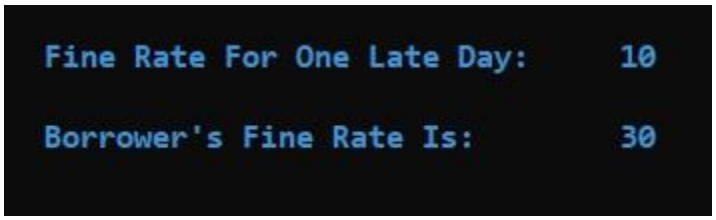
XXII. If you choose update books , you should enter book id and, then you will go to update books menu. So you can choose what you want to update and enter details. Then display a message book details updated

XXIII. If you choose issue books menu from librarian task menu, you will come issue bokks menu. And you can choose show books , issue books ,back to menu or exit. If you choose show books, then you can see all book details . or if you choose issue books you should enter book id , issue details and borrower id.

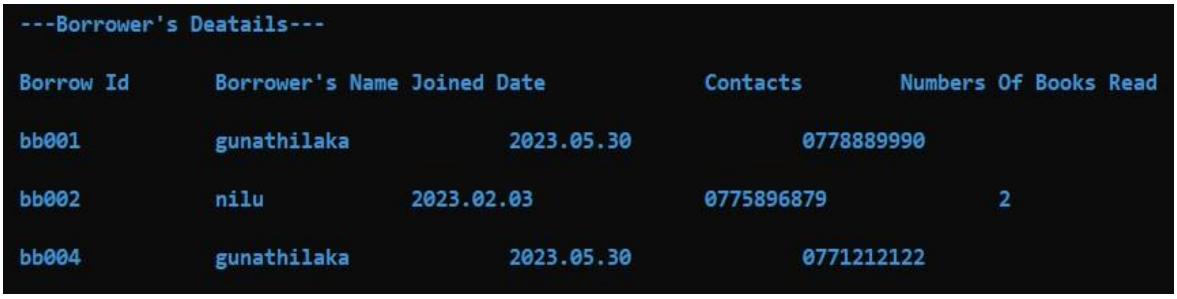


XXIV. Andthen you will display issue file updated and book file updated message and then you can continue or exit

XXV. Then you can go to librarian task menu again. If you choose Accept returned books, you should enter accept book id , returned day and number of late days. Then you can see fine rate



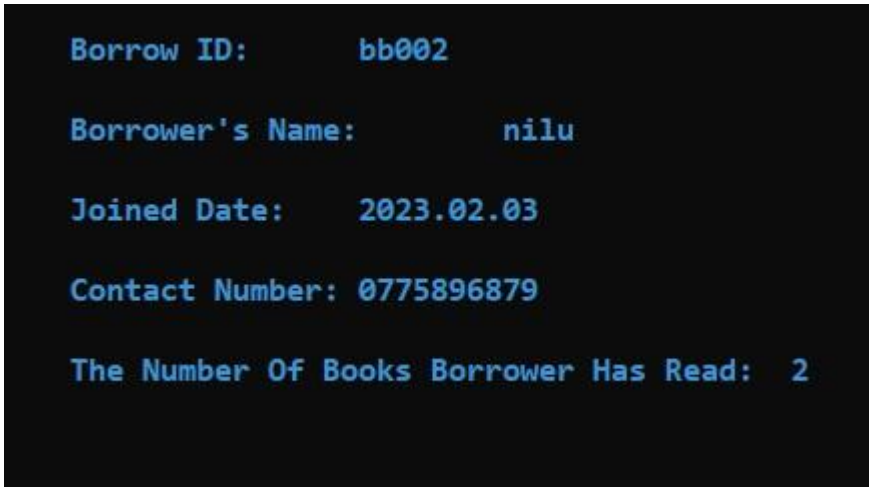
XXVI. If you choose show borrower details from librarian task menu, you can see details of all borrowers



XXVII. If you choose add and delete borrowers you from librarian task menu, you will go to add delete borrowers menu. So you cand delete or add borrowers. If you add borrowers, you should enter borrower details. And after can see message borrower details added

XXVIII. If you want to delete borrower details from borrower file, you should enter borrower id . then you can see borrower details have been deleted message.

XXIX. If you choose search and update borrowers. You can go to search update borrowers menu, and you can choose search or update borrowes. If you choose search borrowers, you should enter borrower id and then you can see all the details of that borrower.



XXX. If you choose update borrowers, you can go to update borrower menu and you can choose what you want to update and then you should enter details for it. Then will display a message borrower details updated.

XXXI. If you want to see who is the currently eligible to be the year end most read prize winner , select show the prize winner. Then you can show all details about that borrower.

08. References.

i. https://www.geeksforgeeks.org/file-handling-c-classes/
ii. https://www.javatpoint.com/cpp-files-and-streams
iii. https://youtu.be/ZzaPdXTrSb8

09. evidence of code implementation with proper comments, indentations, white spaces and meaningful identifiers. (Code Annex)

```
#include <iostream>
#include <bits/stdc++.h>
#include <fstream>
#include <sstream>
#include <string>
#include <windows.h>
#include <cstring> void
issuebooks(); using
namespace std;
void signUpLibrarians(string a, string b) {
        string name;
        string nic;
        string fpassword;
        string fusername;
```

```
string password;
string username;
int count;

password="bsm#"+b;
username="lbr@"+a;

fstream signfile;
signfile.open("librarian.dat",ios::in);
if(!signfile) {
		cout<<"\t\t\t\tFile cannot be open !!!";
	}
else {
	string line;
	while(getline(signfile, line)) {
		istringstream iss(line);
		if(iss>>name>>nic>>fpassword>>fusername) {
			if(name==a && nic==b)
{				count=count+1;

			}
		}
	}
}
signfile.close();

string variable;
signfile.open("librarian.dat",ios::app);
if(!signfile) {
	cout<<"\t\t\t\tFile not open";
}
else {
	if(count==0) {
```

```
            signfile<<a<<","<<b<<","<<username<<","<<password<<endl;
        cout<<endl<<endl<<"\t\t\t\t\tLibrarian's first name:\t"<<a<<endl<<endl;
        cout<<"\t\t\t\t\tLast six digits of NIC:\t"<<b<<endl<<endl;
        cout<<"\t\t\t\t\tLibrarian's username:\t"<<username<<endl<<endl;
        cout<<"\t\t\t\t\tLibrarian's Password:\t"<<password<<endl<<endl;
        cout<<endl<<endl<<"\t\t\t\t\tSignup succesfully"<<endl;
                        }
                        else {
                        cout<<"\t\t\t\t\tThis record has already input! Try again";

                        }
                }
                signfile.close();
        }

        void showlibrianlist() {
                string name;
                string nic;
                string fpassword;
                string fusername;

                cout<<endl<<endl<<"\t\t\t\t\t\t---Librarian Deatails---";
cout<<endl<<endl<<"\t\t\t\t\tLibrarian First Name  "<<"Librarian's User Name
"<<"Librarian's Password";

                fstream librarianFile;
                librarianFile.open("librarian.dat",ios::in);
                if(!librarianFile){
                        cout<<"No such file";
                }
                string line;
```

```
            while (getline(librarianFile, line)) {
                istringstream iss(line);
            getline(iss, name, ',');
            getline(iss, nic, ',');
            getline(iss, fpassword, ',');
                getline(iss, fusername, ',');

cout<<endl<<endl<<"\t\t\t\t\t"<<name<<"\t\t\t"<<fpassword<<"\t\t\t"<<fuserna
me;


            }
            librarianFile.close();

        }

        void ShowFinerateAndMaximumBorrowingLimits() {
            string finerate;       string
        maxximumborrow; fstream
        finerateFile;
            finerateFile.open("finerate.dat",ios::in);
            if(!finerateFile){
                cout<<"No such file";
            }
            string line;
            while (getline(finerateFile, line)) {
            istringstream iss(line);
                getline(iss, finerate, ',');

                cout<<endl<<endl<<"\t\t\t\t\tFine Rate:\t"<<finerate;

            }
            finerateFile.close();
```

```
		fstream maxximumorrowFile;
		maxximumorrowFile.open("maximumborrow.dat",ios::in);
		if(!maxximumorrowFile){
			cout<<endl<<endl<<"\t\t\t\t\tNo such file";
		}
		string line2;
		while (getline(maxximumorrowFile, line2)) {
			istringstream iss(line2);
		getline(iss, maxximumborrow, ',');

cout<<endl<<endl<<"\t\t\t\t\tMaxximum Borrow Limit:\t"<<maxximumborrow;



		}
		maxximumorrowFile.close();

	}
	void addbook() {
		int count;
		string id;
		string bookid;
		string title;
		string author;
		string price;
		string publisher;
		string genre;
		string status;

		string ftitle;
		string fauthor;
		string fprice;
```

```
string fpublisher;
string fgenre;
string fstatus;


cout<<endl<<endl<<"\t\t\t\t\t~~~Add Books~~~"<<endl;

cout<<endl<<endl<<"\t\t\t\t\tEnter Book ID:\t";
cin.ignore();
getline(cin,bookid);

if(system("CLS")) system("clear");
fstream bookfile;
bookfile.open("booklist.dat",ios::in);
if(!bookfile) {
	cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
}
else {
	string line;
	while(getline(bookfile, line)) {
		istringstream iss(line);

			getline(iss, id, ',');
		getline(iss, ftitle, ',');
	getline(iss, fauthor, ',');
	getline(iss, fprice, ',');
			getline(iss, fpublisher, ',');
		getline(iss, fgenre, ',');
			getline(iss, fstatus, ',');
	if(bookid==id){
			count=count+1;
		}
	}
```

```
if(count==0) {
        cout<<endl<<"\t\t\t\t\tEnter Book Title:\t";
        getline(cin,title);

        cout<<endl<<"\t\t\t\t\tEnter Author of Book:\t";
        getline(cin,author);

        cout<<endl<<"\t\t\t\t\tEnter price of the Book:\t";
        getline(cin,price);

        cout<<endl<<"\t\t\t\t\tEnter Publisher of the Book:\t";
        getline(cin,publisher);

        cout<<endl<<"\t\t\t\t\tEnter Genre of the Book:\t";
    getline(cin,genre);

    cout<<endl<<"\t\t\t\t\tEnter Availability status of book:\t";
        getline(cin,status);
        if(system("CLS")) system("clear");

        fstream bookfile;
        bookfile.open("booklist.dat",ios::app);
        if(!bookfile) {
cout<<endl<<endl<<endl<<"\t\t\t\t\tFile can not open"<<endl<<endl;
        }
        else {
            if(id!=bookid) {

    bookfile<<bookid<<","<<title<<","<<author<<","<<price<<","<<publis
her<<","<<genre<<","<<status<<","<<endl;
        cout<<endl<<endl<<"\t\t\t\t\tBook details added"<<endl;

                }
```

```
                        }
                        bookfile.close();
                    }
                    else {
      cout<<endl<<endl<<"\t\t\t\t\tThis book details already added"<<endl;   }
            }
            bookfile.close();
        }

        void deletebook() {
            int count=0;
            string id;
            string bookid;
            string ftitle;
            string fauthor;
            string fprice;
            string fpublisher;
            string fgenre;
            string fstatus;


        cout<<endl<<endl<<"\t\t\t\t\t~~~Delete Books by ID~~~"<<endl<<endl;
cout<<"\t\t\t\t\tEnter Book Id:\t"; cin>>id;
            if(system("CLS")) system("clear");

            fstream bookfile;
            bookfile.open("booklist.dat",ios::in);

            fstream tempfile;
            tempfile.open("temp.dat",ios::out);

            string line;
            while (getline(bookfile, line)) {
```

```
            istringstream iss(line);

                        getline(iss, bookid, ',');
                    getline(iss, ftitle, ',');
                getline(iss, fauthor, ',');
            getline(iss, fprice, ',');
            getline(iss, fpublisher, ',');
            getline(iss, fgenre, ',');
            getline(iss, fstatus, ',');
            if(bookid!=id){

            tempfile<<bookid<<","<<ftitle<<","<<fauthor<<","<<fprice<<","<<fpu
blisher<<","<<fgenre<<","<<fstatus<<","<<endl;

                    }
                    else if(bookid==id){
                        count=1;
                    }       }
            if(count!=1) {
                cout<<endl<<endl<<"\t\t\t\t\tThis Record Was Not Found!!";
            } else
            {

            cout
            <<en
            dl<<
            "\t\t
            \t\t\
            tThe
            Book
            Deta
            ils
            Have
            Been
```

```
            Dele
            ted";
            }
            bookfile.close();
            tempfile.close();

            remove("booklist.dat");
            rename("temp.dat","booklist.dat");
        }

        void searchbook() {
            int flag=0;
            string id;
            string ftitle;
            string fauthor;
            string fprice;
            string fpublisher;
            string fgenre;
            string fstatus;

            string bookidORtitle;


        cout<<endl<<endl<<"\t\t\t\t\tSearch Books By Book Name Or Book
ID"<<endl<<endl;  cout<<endl<<"\t\t\t\t\tEnter Name Of The Book Or Book
        ID:\t";         cin>>bookidORtitle;

            fstream bookfile;
            bookfile.open("booklist.dat",ios::in);
            if(!bookfile) {
                cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
            } else { string
            line;
                while(getline(bookfile, line)) {
```

```
                istringstream iss(line);

                getline(iss, id, ',');
                getline(iss, ftitle, ',');
            getline(iss, fauthor, ',');
            getline(iss, fprice, ',');
                getline(iss, fpublisher, ',');
            getline(iss, fgenre, ',');
                getline(iss, fstatus, ',');

                        if(bookidORtitle==id || bookidORtitle==ftitle) {
        cout<<endl<<endl<<"\t\t\t\t\tID Of The Book:\t"<<id<<endl<<endl;
        cout<<"\t\t\t\t\tName Of The Book:\t"<<ftitle<<endl<<endl;
        cout<<"\t\t\t\t\tAuthor Of The Book:\t"<<fauthor<<endl<<endl;
        cout<<"\t\t\t\t\tPrice Of The Book:\t"<<fprice<<endl<<endl;
        cout<<"\t\t\t\t\tPublisher Of The Book:\t"<<fpublisher<<endl<<endl;
        cout<<"\t\t\t\t\tGenre Of The Book:\t"<<fgenre<<endl<<endl;
        cout<<"\t\t\t\t\tThis Book is "<<fstatus<<" now"<<endl<<endl;
                            flag=1;
                        }
                }
        }
        bookfile.close();
        if(flag!=1) {
        cout<<endl<<endl<<"\t\t\t\t\tThis Record Was Not Found!!"<<endl;
        }
}

void updatebook() {
        string title;
        string fauthor;
        string author;
        string fprice;
        string price;
```

```
        string fpublisher;
        string publisher;
        string fgenre;
        string genre;
        string fstatus;
        string status; string
        bookname;
        string bookidORtitle;
        int option;    int
flag=1;         int count=0;
        int mark=0;  string
id;     string bookid;
        string ftitle;

        cout<<endl<<endl<<"\t\t\t\t\t~~~Update Books~~~"<<endl<<endl;
        cout<<"\t\t\t\t\tUpdate Books By Book ID"<<endl<<endl;
        cout<<"\t\t\t\t\tEnter Book ID:\t";
        cin>>bookidORtitle;
        if(system("CLS")) system("clear");

        fstream newFile;
        newFile.open("booklist.dat",ios::in);
        if(!newFile){
        cout<<"No such file";
        }
        string line;
        while (getline(newFile, line)) {
                istringstream iss(line);
                getline(iss, bookid, ',');
                getline(iss, title, ',');
        getline(iss, author, ',');
                                getline(iss, price, ','); getline(iss,
                                publisher, ','); getline(iss, genre,
                                ','); getline(iss, status, ',');
```

```
                        if(bookidORtitle==bookid) {
                                mark=1;
                        }

        }
        newFile.close();

                        if(mark==1) {

                                while(flag==1) {
cout<<endl<<"\t\t\t\t\tWhat Do You Want To Update?"<<endl<<endl;
cout<<"\t\t\t\t\t01. Name Of The Book"<<endl<<endl;
cout<<"\t\t\t\t\t02. Author Of The Book"<<endl<<endl;
cout<<"\t\t\t\t\t03. Price Of The Book"<<endl<<endl;
cout<<"\t\t\t\t\t04. Publisher Of The Book"<<endl<<endl;
cout<<"\t\t\t\t\t05. Genre Of The Book"<<endl<<endl;
cout<<"\t\t\t\t\t06. Availability status"<<endl<<endl;
cout<<"\t\t\t\t\t07. Back to Menu"<<endl<<endl; cout<<"\t\t\t\t\tPleace
Choose Any Option: ";
                                cin>>option;
                                if(system("CLS")) system("clear");

                                switch(option) {
                                        case 1:{

        cout<<endl<<endl<<endl<<"\t\t\t\t\tEnter Name Of The Book:\t";
                                                cin.ignore();
                                getline(cin,title);
                                        if(system("CLS")) system("clear");
                        flag=1;
                                                break;
                                        }
                                        case 2:{
```

```
        cout<<endl<<endl<<endl<<"\t\t\t\t\tEnter The Author Of The
Book:\t";
                                                cin.ignore();
                                                getline(cin,author);
                                        if(system("CLS")) system("clear");
                                                flag=1;
                                                break;
                                        }
                                        case 3:{

        cout<<endl<<endl<<endl<<"\t\t\t\t\tEnter Price Of The Book:\t";
                                                cin.ignore();
                                    getline(cin,price);
                                                if(system("CLS")) system("clear");
                                                flag=1;
                                                break;
                                        }
                                        case 4:{

        cout<<endl<<endl<<endl<<"\t\t\t\t\tEnter Publisher Of The Book:\t";
                                                cin.ignore();
                                                getline(cin,publisher);
                                        if(system("CLS")) system("clear");
                                                flag=1;
                                                break;
                                        }
                                        case 5:{

        cout<<endl<<endl<<endl<<"\t\t\t\t\tEnter The Genre Of The
Book:\t";
                                                cin.ignore();
                                                getline(cin,genre);
```

```
                                        if(system("CLS")) system("clear");
                                                flag=1;
                                                break;
                        }               case 6:{

 cout<<endl<<endl<<endl<<"\t\t\t\t\tEnter The Availability status Of The
Book:\t";
                                                cin.ignore();
                                getline(cin,status);
                                        if(system("CLS")) system("clear");
                                                flag=1;
                                                break;
                                        }
                                        case 7:{
                                        flag=0;
                                                break;
                                        }
                                        default :{

        cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter!!!"<<endl;
                                                flag=1;
                                                break;
                                        }
                                }

                                }
        fstream bookfile2;
        bookfile2.open("booklist.dat",ios::in);

        fstream tempfile2;
        tempfile2.open("temp.dat",ios::out);

        string line2;
```

```
        while (getline(bookfile2, line2)) { istringstream
                iss(line2); getline(iss, bookid, ',');
                                getline(iss, ftitle, ',');
                        getline(iss, fauthor, ',');
                                getline(iss, fprice, ',');
                getline(iss, fpublisher, ',');
        getline(iss, fgenre, ',');
                                getline(iss, fstatus, ',');

                        if(bookidORtitle!=bookid){

        tempfile2<<bookid<<","<<ftitle<<","<<fauthor<<","<<fprice<<","<<fp
ublisher<<","<<fgenre<<","<<fstatus<<","<<endl;
                        }
                        else if(bookidORtitle==bookid){

        tempfile2<<bookid<<","<<title<<","<<author<<","<<price<<","<<publ
isher<<","<<genre<<","<<status<<","<<endl;
                                count=1;
                        }
        }
        if(count==1) {
                cout<<endl<<"\t\t\t\t\tThe Book Details Have Been Updated";
        }
        bookfile2.close();
        tempfile2.close();

        remove("booklist.dat");
        rename("temp.dat","booklist.dat");

        }
        else {
                cout<<endl<<endl<<"\t\t\t\t\tThis Record Was Not Found!!";
```

```
		}

	}
	void showbooks() {
		string bookid;
		string id;
		string ftitle;
		string fauthor;
		string fprice;
		string fpublisher;
		string fgenre;
		string fstatus;

		cout<<endl<<endl<<"\t\t\t\t\t,,,,,,,,,____BOOK LIST____,,,,,,,,,";
cout<<endl<<endl<<"\tID Of The Book"<<"\tName Of The Book"<<"\tAuthor Of The Book"<<"\tGenre Of The Book"<<"\tAvailability status"<<endl<<endl;

		fstream bookfile;    //show book list
		bookfile.open("booklist.dat",ios::in);
		if(!bookfile) {
			cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
		}
		else {
			string line;
			while(getline(bookfile, line)) {
				istringstream iss(line);

				getline(iss, id, ',');
			getline(iss, ftitle, ',');
		getline(iss, fauthor, ',');
		getline(iss, fprice, ',');
		getline(iss, fpublisher, ',');
```

```
                getline(iss, fgenre, ',');
                getline(iss, fstatus, ',');
            cout<<"\t"<<id<<"\t\t"<<ftitle<<"\t\t"<
            <fauthor<<"\t\t"<<fgenre<<"\
t\t"<<fstatus<<endl<<endl;
                }
        }
        bookfile.close();
}

void showborrowerdeatails() {
        string id;
        string fname;
        string fdate;
        string fcontact;
        string freadbooks;

        cout<<endl<<endl<<"\t\t\t\t---Borrower's Deatails---";
cout<<endl<<endl<<"\t\t\t\tBorrow Id\t"<<"Borrower's Name\t"<<"Joined
Date\t\t"<<"Contacts\t"<<"Numbers Of Books Read";

        fstream borrowfile;
        borrowfile.open("borrowfile.dat",ios::in);
        if(!borrowfile) {
            cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
        }
        else {
            string line;
            while(getline(borrowfile, line)) {
                istringstream iss(line);

                    getline(iss, id, ',');
            getline(iss, fname, ',');
```

```
                getline(iss, fdate, ',');
                getline(iss, fcontact, ',');
                getline(iss, freadbooks, ',');
                cout<<endl<<endl<<"\t\t\t\t"<<id<
                <"\t\t"<<fname<<"\t\t"<<fdate<<
"\t\t"<<fcontact<<"\t\t"<<freadbooks;


        }
}
borrowfile.close();
}

void issuebooks() {
        int smark;    int mark;
        int mark2;    int count=1;
        int flag=1;    int
Maxximumborrow;        int
value=0;      string
maxximumborrow;        string
borrowid;    string issuedate;
        string returndate;
        string bookid;        string
id;      string ftitle;  string
fauthor;       string fprice;
        string fpublisher;    string
fgenre;
        string fstatus;
        string status; string
        FineRate; string
        returnedDay;
        //read maxximum
        borrow file to get
        maxximum borrow
```

```
value. fstream
maxximumborrowf
ile;
maxximumborrowf
ile.open("maximu
mborrow.dat",ios::i
n);
if(!maxximumborrowfile) {
        cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
}
else {
        string line;
        while(getline(maxximumborrowfile, line)) {
istringstream iss(line);
if(iss>>maxximumborrow) {
                        cout<<endl<<endl<<"\t\t\t\t\tMaxximum Borrow
Limit:\t"<<maxximumborrow<<endl;
        }
                else {
                        cout<<endl<<endl<<"\t\t\t\t\tCan't read"    ;
                }
        }
}
maxximumborrowfile.close();

stringstream ss(maxximumborrow);
ss>>Maxximumborrow;

while(flag==1) {
cout<<endl<<endl<<"\t\t\t\t\tEnter The Book ID To Be Issued:\t";
cin>>bookid;
        if(system("CLS")) system("clear");
```

```
        fstream bookfile;
        bookfile.open("booklist.dat",ios::in);
        if(!bookfile) {
                cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
        } else
        {
        strin
        g
        line;
                while(getline(bookfile, line)) {
                        istringstream iss(line);

                        getline(iss, id, ',');
                        getline(iss, ftitle, ',');
                        getline(iss, fauthor, ',');
        getline(iss, fprice, ',');
getline(iss, fpublisher, ',');
getline(iss, fgenre, ',');
                        getline(iss, fstatus, ',');

                        if(bookid==id) {
                                mark=1;
                                if(fstatus=="available") {
                                        smark=1;
                                }
                        }

                }
        }
        bookfile.close();

        if(mark!=1) {
                cout<<endl<<endl<<"\t\t\t\t\tInvalid Book ID";
```

```
            }
            if(mark==1) {

                  if(count<=Maxximumborrow && smark==1) {

                  cout<<endl<<endl<<"\t\t\t\t\tEnter Borrower ID:\t";
                  cin.ignore();
      getline(cin,borrowid);
       cout<<endl<<endl<<"\t\t\t\t\tEnter Issue Date(yyyy.mm.dd):\t";
                        getline(cin,issuedate);
      cout<<endl<<endl<<"\t\t\t\t\tEnter Due Data(yyyy.mm.dd):\t";
                        getline(cin,returendate);
                        if(system("CLS")) system("clear");

                        fstream issuefile;
                        issuefile.open("issue.dat",ios::app);
                        if(!issuefile) {
                        cout<<endl<<endl<<"\t\t\t\t\tFile Can Not Open";
                        }
                        else {

      issuefile<<borrowid<<","<<bookid<<","<<issuedate<<","<<returendat
e<<","<<FineRate<<","<<returnedDay<<endl;
                        cout<<endl<<endl<<"\t\t\t\t\tIssue File Updated...";
                              count++;
                        }
                        issuefile.close();
                        string fname;
                        string fdate;
                        string fcontact;
                        string freadbooks;
                        string nofbookstring;
```

```
                fstream borrowfile;
                borrowfile.open("borrowfile.dat",ios::in);
        if(!borrowfile) {
                        cout<<endl<<endl<<"\t\t\t\t\tFile cannot
be open !!!";
                }
                else {
                        string line;
                        while(getline(borrowfile, line)) {
                        istringstream iss(line);

                                getline(iss, id, ',');
                                getline(iss, fname, ',');
                                getline(iss, fdate, ','); getline(iss,
                                fcontact, ',');
                                getline(iss, freadbooks, ',');

                                if(borrowid==id && count>=1){
                                        value=1;
                                }
                        }
                }
                        borrowfile.close();
                        if(value==1) {
                        int nofbooks;
                        stringstream ss(freadbooks);
                ss>>nofbooks;   //convert nomber of read books to int

                                nofbooks=nofbooks+1;
                        // again convert to string
                        stringstream stream;
                stream << nofbooks;
                                stream >> nofbookstring;
                        }
```

```
string bid , bfname , bfdate , bfcontact , bfreadbooks;

borrowfile.open("borrowfile.dat",ios::in);

fstream tempfile;
tempfile.open("temp.dat",ios::out);

string line2;
while (getline(borrowfile, line2)) {
    istringstream iss(line2);

    getline(iss, bid, ',');
    getline(iss, bfname, ',');
    getline(iss, bfdate, ','); getline(iss,
    bfcontact, ',');
    getline(iss, bfreadbooks, ',');


        if(borrowid!=bid){

tempfile<<bid<<","<<bfname<<","<<bfdate<<","<<bfcontact<<","<<b
freadbooks<<endl;                                                    }
                                    else if(borrowid==bid){


tempfile<<bid<<","<<bfname<<","<<bfdate<<","<<bfcontact<<","<<n
ofbookstring<<endl;
                                            mark2=1;

                                    }

}
```

```
borrowfile.close();
tempfile.close();

remove("borrowfile.dat");
rename("temp.dat","borrowfile.dat");

fstream bookfile;
        bookfile.open("booklist.dat",ios::in);

        tempfile.open("temp.dat",ios::out);

        while (getline(bookfile, line2)) {
                istringstream iss(line2);

                        getline(iss, id, ',');
                getline(iss, ftitle, ',');
                getline(iss, fauthor, ',');
                getline(iss, fprice, ',');
                getline(iss, fpublisher, ',');
                        getline(iss, fgenre, ',');
                        getline(iss, status, ',');


                if(bookid!=id){

        tempfile<<id<<","<<ftitle<<","<<fauthor<<","<<fprice<<","<<fpublish
er<<","<<fgenre<<","<<status<<endl;
                }
                else if(bookid==id){
                        fstatus="unavailable";

        tempfile<<id<<","<<ftitle<<","<<fauthor<<","<<fprice<<","<<fpublish
er<<","<<fgenre<<","<<"unavailable"<<endl;
```

```
                                        mark2=1;
                                    }

                                }
                        bookfile.close();
                        tempfile.close();

                        remove("booklist.dat");
                        rename("temp.dat","booklist.dat");

                        if(mark2==1) {
                                cout<<endl<<endl<<"\t\t\t\t\tBook List
Updated";           }
                } else {
                        if(co
                        unt>
                        Max
                        ximu
                        mbo
                        rrow
                        ) {
cout<<endl<<endl<<"\t\t\t\t\tYou Can Issue Only "<<Maxximumborrow<<"
Books"<<endl;
                }
                                else if(smark!=1) {
        cout<<endl<<endl<<"\t\t\t\t\tThis Book Is Unavailable Now"<<endl;

                                }


                        }
```

```
                cout<<endl<<endl<<"\t\t\t\t\tDo You Want To";
        cout<<endl<<endl<<"\t\t\t\t\t01.Continue";
        cout<<endl<<endl<<"\t\t\t\t\t02.Exit";
                cout<<endl<<endl<<"\t\t\t\t\tCoose A Option: ";
                cin>>flag;
                if(system("CLS")) system("clear");

                if(flag==2) {
                        count=0;
                        break;
                }
            }

        }

}

void acceptReturnedBooks(){
        string returnedDay;
        string issueBookId;
        string borrowID;
        string borrowid;
        string bookid;
        string issuedate;
        string returendate;
        string finerate; int
        flag=1; int latedays;
        string FineRate;
        string ReturnedDay;
        int fineRate;
        int mark2;
        string id;
        string ftitle;
        string fauthor;
```

```
        string fprice;
        string fpublisher;
        string fgenre;
        string status;
        string fineRateString;

        while(flag==1) {
                cout<<endl<<endl<<"\t\t\t\t\tEnter Book ID:\t";
        cin.ignore();
                getline(cin,issueBookId);

                cout<<endl<<endl<<"\t\t\t\t\tEnter Returned Day
(yyyy/mm/dd):\t";
                getline(cin,ReturnedDay);

                cout<<endl<<endl<<"\t\t\t\t\tEnter Borrow ID:\t";
                getline(cin,borrowID);
                if(system("CLS")) system("clear");

                        //open issue details file to get returen date and other
                values. fstream issuefile;
                issuefile.open("issue.dat",ios::in);
                if(!issuefile) { cout<<endl<<endl<<"\t\t\t\t\tFile cannot be
                        open !!!";
                }
                else {
                        string line;
                        while(getline(issuefile, line)) {
                        istringstream iss(line);
                                getline(iss, borrowid, ',');
                                getline(iss, bookid, ',');
                getline(iss, issuedate, ',');
```

```
                getline(iss, returendate, ',');
                getline(iss, FineRate, ',');
                                getline(iss, returnedDay, ',');

                                if(issueBookId==bookid) {
cout<<endl<<endl<<"\t\t\t\t\tDue
Date(yyyy.mm.dd):\t"<<returendate<<endl<<endl;
                                }

                        }
                }
                issuefile.close();

                //read fine rate file to get fine rate value.

                fstream fineratefile;
                fineratefile.open("finerate.dat",ios::in);
                if(!fineratefile) {
                        cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
                }
                else {
                        string line;
                        while(getline(fineratefile, line)) {
                                istringstream iss(line);
                                if(iss>>finerate) {
                                }
                                else {
                                        cout<<endl<<endl<<"\t\t\t\t\tCan't read";
                                }
                        }
                }
                fineratefile.close();
```

```
		//calculate fine rate

		int FINErate;

		stringstream ss(finerate);
		ss>>FINErate;

		cout<<endl<<endl<<"\t\t\t\t\tEnter Number Of Days Late:\t";
	cin>>latedays;
		if(system("CLS")) system("clear");
	cout<<endl<<endl<<"\t\t\t\t\tFine Rate For One Late Day:\t"<<finerate;

		fineRate=FINErate*latedays;

cout<<endl<<endl<<"\t\t\t\t\tBorrower's Fine Rate Is:\t"<<fineRate<<endl<<endl;

		stringstream stream;
	stream << fineRate;
		stream >> fineRateString;

			issuefile.open("issue.dat",ios::in);

	fstream tempfile;
			tempfile.open("temp2.dat",ios::out);

			string line2;
			while (getline(issuefile, line2)) {
				istringstream iss(line2); getline(iss,
				borrowid, ',');
				getline(iss, bookid, ',');
				getline(iss, issuedate, ',');
				getline(iss, returendate, ',');
				getline(iss, FineRate, ',');
```

```
                    getline(iss, returnedDay, ',');

                    if(issueBookId!=bookid){

        tempfile<<borrowid<<","<<bookid<<","<<issuedate<<","<<returnda
te<<","<<FineRate<<","<<returnedDay<<endl;
                    }
                    else if(issueBookId==bookid){


        tempfile<<borrowid<<","<<bookid<<","<<issuedate<<","<<returnda
te<<","<<fineRateString<<","<<ReturnedDay<<endl;
                    }
                }


                    issuefile.close();
                    tempfile.close();

                    remove("issue.dat");
                    rename("temp2.dat","issue.dat");


                    fstream bookfile;
                            bookfile.open("booklist.dat",ios::in);

                            fstream tempfile2;
                            tempfile2.open("temp.dat",ios::out);

                            string line3;
                            while (getline(bookfile, line3)) {
                                istringstream iss(line3);
```

```
                                        getline(iss, id, ',');
                                        getline(iss, ftitle, ',');
                                    getline(iss, fauthor, ',');
                                getline(iss, fprice, ',');
                            getline(iss, fpublisher, ',');
                            getline(iss, fgenre, ',');
                                        getline(iss, status, ',');


                                        if(issueBookId!=id){

        tempfile2<<id<<","<<ftitle<<","<<fauthor<<","<<fprice<<","<<fpublis
her<<","<<fgenre<<","<<status<<endl;
                        }
                                        else if(issueBookId==id){


        tempfile2<<id<<","<<ftitle<<","<<fauthor<<","<<fprice<<","<<fpublis
her<<","<<fgenre<<","<<"available"<<endl;
                                                mark2=1;
                                        }

                                    }
                                bookfile.close();
                                tempfile2.close();

                                remove("booklist.dat");
                                rename("temp.dat","booklist.dat");

                            cout<<endl<<endl<<"\t\t\t\t\tDo You Want To";
                            cout<<endl<<endl<<"\t\t\t\t\t01. Contineue";
                            cout<<endl<<endl<<"\t\t\t\t\t02. Exit";
                    cout<<endl<<endl<<"\t\t\t\t\tCoose Any Option: ";
                                cin>>flag;
```

```
                                        if(system("CLS")) system("clear");
                                        if(flag==2) {
                                                break;
                                        }

                        }
}
void addborrowers(){
        string borrowid;
        string name;
        string date;
        string contact;
        string id;
        string fname;
        string fdate;
        string fcontact;
        string freadbooks;
        int count=0;

        cout<<endl<<endl<<"\t\t\t\t\t~~~Add Borrowers~~~"<<endl;

        cout<<endl<<endl<<"\t\t\t\t\tEnter Borrower ID:\t";
        cin.ignore();
        getline(cin,borrowid);

        if(system("CLS")) system("clear");
        fstream borrowfile;
        borrowfile.open("borrowfile.dat",ios::in);
        if(!borrowfile) {
                cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
        } else
        {
        strin
```

```
g
line;
        while(getline(borrowfile, line)) {
                istringstream iss(line);

                        getline(iss, id, ',');
                        getline(iss, fname, ',');
                        getline(iss, fdate, ',');
        getline(iss, fcontact, ',');
                        getline(iss, freadbooks, ',');

                        if(id==borrowid) {
                                count=count+1;
                        }

        }
        if(count==0) {

                cout<<endl<<"\t\t\t\t\tEnter Borrower Name\t:";
    getline(cin,name);

        cout<<endl<<"\t\t\t\t\tEnter Joined Date(yyyy.mm.dd):\t";
                getline(cin,date);

                cout<<endl<<"\t\t\t\t\tEnter Contact Number:\t";
                getline(cin,contact);
                if(system("CLS")) system("clear");

                fstream borrowfile;
                borrowfile.open("borrowfile.dat",ios::app);
                if(!borrowfile) {
cout<<endl<<endl<<endl<<"\t\t\t\t\tFile can not open"<<endl<<endl;
                }
```

```
                    else {
                    if(borrowid!=id) {
                borrowfile<<borrowid<<","<<
                name<<","<<date<<","<<cont
                act<<","<
<freadbooks<<endl;
        cout<<endl<<endl<<"\t\t\t\t\tBorrower's details added"<<endl;

                            }
                    }
                    borrowfile.close();
                }
        else {
cout<<endl<<endl<<"\t\t\t\t\tThe Borrower Details Already Added"<<endl;

                }

        }

        borrowfile.close();

}


void deleteBorrowers() {
        string borrowid;
        string id;
        string fname;
        string fdate;
        string fcontact;
        string freadbooks;
        int count=0;
```

```
cout<<endl<<endl<<"\t\t\t\t\t~~~Delete Books by ID~~~"<<endl<<endl;
	cout<<"\t\t\t\t\tEnter Borrow Id:\t";
	cin>>borrowid;
	if(system("CLS")) system("clear");

	fstream borrowfile;
	borrowfile.open("borrowfile.dat",ios::in);

	fstream tempfile;
	tempfile.open("temp.dat",ios::out);

	string line;
	while (getline(borrowfile, line)) {
		istringstream iss(line);

				getline(iss, id, ',');
			getline(iss, fname, ',');
		getline(iss, fdate, ',');
	getline(iss, fcontact, ',');
				getline(iss, freadbooks, ',');
		if(borrowid!=id){

	tempfile<<id<<","<<fname<<","<<fdate<<","<<fcontact<<","<<freadb
ooks<<endl;
			}
			else if(borrowid==id){
				count=1;
			}	}
	if(count!=1) {
		cout<<endl<<endl<<"\t\t\t\t\tThis Record Was Not Found!!";
	}
	else {
		cout<<endl<<"\t\t\t\t\tThe Book Details Have Been Deleted";
```

```
        }
        borrowfile.close();
        tempfile.close();

        remove("borrowfile.dat");
        rename("temp.dat","borrowfile.dat");
}

void searchBorrowers() {
        string borrowidORname;
        string id;      string
fname;        string fdate;
        string fcontact;      string
freadbooks;
        int flag=0;

        cout<<endl<<endl<<"\t\t\t\t\tSearch Borrowers By Borrower's Name
Or Borrower's ID"<<endl<<endl;
cout<<endl<<"\t\t\t\t\tEnter Name Of The Borrower Or Borrow ID:\t";
        cin>>borrowidORname;
        if(system("CLS")) system("clear");

        fstream borrowfile;
        borrowfile.open("borrowfile.dat",ios::in);
        if(!borrowfile) {
                cout<<endl<<endl<<"\t\t\t\t\tFile cannot be open !!!";
        }
        else {
                string line;
                while(getline(borrowfile, line)) {
                        istringstream iss(line);

                                getline(iss, id, ',');
                        getline(iss, fname, ',');
```

```
                getline(iss, fdate, ',');
        getline(iss, fcontact, ',');
                                getline(iss, freadbooks, ',');

                                if(borrowidORname==id || borrowidORname==fname) {
                        cout<<endl<<endl<<"\t\t\t\t\tBorrow ID:\t"<<id<<endl<<endl;
                        cout<<"\t\t\t\t\tBorrower's Name:\t"<<fname<<endl<<endl;
                        cout<<"\t\t\t\t\tJoined Date:\t"<<fdate<<endl<<endl;
                        cout<<"\t\t\t\t\tContact Number:\t"<<fcontact<<endl<<endl;
                        cout<<"\t\t\t\t\tThe Number Of Books Borrower Has
Read:\t"<<freadbooks<<endl<<endl;

                                                flag=1;
                                        }
                                }
        }
        borrowfile.close();
        if(flag!=1) {
        cout<<endl<<endl<<"\t\t\t\t\tThis Record Was Not Found!!"<<endl;
        }
}

void UpdateBorrowers() {

        string borrowid;
        string id;
        string fname;
        string fdate;
        string fcontact;
        string freadbooks;
        string name;
        string date;
        string contact;
```

```
		string readbooks;
		int flag=1;
		int count=0;
		int option;
		int mark=0;

cout<<endl<<endl<<"\t\t\t\t\t~~~Update Borrowers' Deatails~~~"<<endl<<endl;
cout<<"\t\t\t\t\tUpdate Borrower Deatails By Borrower ID"<<endl<<endl;
	cout<<"\t\t\t\t\tEnter Borrow ID: ";
		cin>>borrowid;
		if(system("CLS")) system("clear");

		fstream newFile;
		newFile.open("borrowfile.dat",ios::in);
		if(!newFile){
			cout<<endl<<endl<<endl<<"\t\t\t\t\tNo such file";
		}
		string line;
		while (getline(newFile, line)) {
		istringstream iss(line);
		getline(iss, id, ',');
		getline(iss, fname, ',');
		getline(iss, fdate, ',');
		getline(iss, fcontact, ',');
					getline(iss, freadbooks, ',');

					if(borrowid==id) {
						mark=1;
					}

		}
		newFile.close();
```

```
if(mark==1) {

while(flag==1) {
cout<<endl<<"\t\t\t\t\tWhat Do You Want To Update?"<<endl<<endl;
cout<<"\t\t\t\t\t01. Borrow's Name"<<endl<<endl;
cout<<"\t\t\t\t\t02. Joined Date"<<endl<<endl; cout<<"\t\t\t\t\t03.
Borrower's Contacts"<<endl<<endl; cout<<"\t\t\t\t\t04. Back to
Menu"<<endl<<endl;
cout<<"\t\t\t\t\tPleace Choose Any Option: ";
cin>>option;
if(system("CLS")) system("clear");

switch(option) {
case 1:{

cout<<endl<<endl<<endl<<"\t\t\t\t\tEnter Borrow's Name:\t";
cin.ignore();
getline(cin,fname);
if(system("CLS")) system("clear");
flag=1;
break;
}
case 2:{

cout<<endl<<endl<<endl<<"\t\t\t\t\tEnter Joined Date:\t";
cin.ignore();
getline(cin,fdate);
if(system("CLS")) system("clear");
flag=1;
break;
}
case 3:{
```

```
cout<<endl<<endl<<endl<<"\t\t\t\t\tBorrower's Contacts:\t";
                                cin.ignore();
                                        getline(cin,fcontact);
                                    if(system("CLS")) system("clear");
                                        flag=1;
                                        break;
                                    }
                                    case 4:{
                                    flag=0;
                                cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter!!!"<<endl;
                                        flag=1;
                                        break;
                                    }
                                }

                            }
fstream borrowfile;
borrowfile.open("borrowfile.dat",ios::in);

fstream tempfile2;
tempfile2.open("temp.dat",ios::out);

string line2;
while (getline(borrowfile, line2)) {
        istringstream iss(line2);

                    getline(iss, id, ',');
                getline(iss, name, ',');
```

```
                getline(iss, date, ',');
        getline(iss, contact, ',');
                                getline(iss, readbooks, ',');

                                if(borrowid!=id){

 tempfile2<<id<<","<<name<<","<<date<<","<<contact<<","<<readbo
oks<<endl;

                                }
                                else if(borrowid==id){
        tempfile2<<id<<","<<fname<<","<<fdate<<","<<fcontact<<","<<fread
books<<endl;
                                count=1;

                                }
        }
        if(count==1) {
                cout<<endl<<"\t\t\t\t\tThe Borrower Details Have Been
Updated";
        }

        borrowfile.close();
        tempfile2.close();

        remove("borrowfile.dat");
        rename("temp.dat","borrowfile.dat");
 }
else {
        cout<<endl<<endl<<"\t\t\t\t\tThis Record Was Not Found ";
}

}
```

```
void prizewinner() {

        int largestValue , nofbooks;
        string id , fname , fdate , fcontact , freadbooks , nofbookstring;


cout<<endl<<endl<<"\t\t\t\t\t***Borrower, The Prize Winner Who Has Read The
Most Books***";


        fstream newFile;
        newFile.open("borrowfile.dat",ios::in);
        if(!newFile){
                cout<<endl<<endl<<endl<<"\t\t\t\t\tNo such file";
        }
        string line;
        while (getline(newFile, line)) {
                istringstream iss(line);
        getline(iss, id, ',');
        getline(iss, fname, ',');
        getline(iss, fdate, ',');
        getline(iss, fcontact, ',');
                getline(iss, freadbooks, ',');


                stringstream ss(freadbooks);
                ss>>nofbooks;   //convert nomber of read books to int


                if(largestValue<nofbooks) {
                        largestValue=nofbooks;
                }

        }
```

```
        newFile.close();

        // again convert to string
                stringstream stream;
        stream << nofbooks;
                stream >> nofbookstring;

        newFile.open("borrowfile.dat",ios::in);
        if(!newFile){
                cout<<endl<<endl<<endl<<"\t\t\t\t\tNo such file";
        }
        while (getline(newFile, line)) {
                istringstream iss(line);
                getline(iss, id, ',');
                getline(iss, fname, ',');
                getline(iss, fdate, ',');
                getline(iss, fcontact, ',');
                getline(iss, freadbooks, ',');

                if(freadbooks==nofbookstring) {
        cout<<endl<<endl<<"\t\t\t\t\tBorrower ID:\t"<<id;
        cout<<endl<<endl<<"\t\t\t\t\tBorrower Name:\t"<<fname;
        cout<<endl<<endl<<"\t\t\t\t\tBorrower Contact:\t"<<fcontact;
cout<<endl<<endl<<"\t\t\t\t\tNumbe Of Books Read:\t"<<freadbooks;
                }
        }
        newFile.close();

        int backtomenu;
        cout<<endl<<endl<<"\t\t\t\t\tEnter Any key To Back To Menu:\t";
        cin>>backtomenu;
        if(system("CLS")) system("clear");
}
```

```
int main(int argc, char** argv) {
	HANDLE h= GetStdHandle(STD_OUTPUT_HANDLE);
	static CONSOLE_FONT_INFOEX fontex;
	fontex.cbSize = sizeof(CONSOLE_FONT_INFOEX);
	HANDLE hOut = GetStdHandle(STD_OUTPUT_HANDLE);
	GetCurrentConsoleFontEx(hOut, 0, &fontex);
	fontex.FontWeight = 700;	fontex.dwFontSize.X =
20;	fontex.dwFontSize.Y = 20;
	SetCurrentConsoleFontEx(hOut, NULL, &fontex);

	int menu3;
	int menu2=1; int
	menu1=1;
	int length; int
	addDeleteBorrowers; int
	searchUpdateBorrowers;
	string adminPassword;
	string adminPasswordOriginal="1234";
	string Lusername;
	string Lpassword;
	string username;
	string password;
	string lnic;	string
lname;	string name;
	string nic;	string
fpassword;	string
fusername;	int
searchORupdate=1;	int
fineRate=10;	int
maximumLimit=2;
	int issuebooks=1;

		while(menu1<=2) {
		SetConsoleTextAttribute(h,3);
```

```
        cout<<endl<<endl<<endl<<"\t\t\t\t_____
_____"<<endl<<endl;
cout<<"\t\t\t\t       WELCOME TO BLOSSOM LIBRARY SYSTEM       "<<endl;

        cout<<"\t\t\t\t_____
____"<<endl<<endl;              cout<<"\t\t\t\t\t01.
        Admin"<<endl<<endl;     cout<<"\t\t\t\t\t02.
        Librarian"<<endl<<endl; cout<<"\t\t\t\t\t03. Close
        Application"<<endl; cout<<endl<<"\t\t\t\t\tPleace Choose Any
        Option to login: "; cin>>menu1;
                if(system("CLS")) system("clear");


                        if(menu1==1) {
                                cout<<endl<<"\t\t\t\t\tEnter Admin password:\t";
                                cin>>adminPassword;
                cout<<endl;
                                if(system("CLS")) system("clear");
                                if(adminPassword==adminPasswordOriginal) {
                                        menu2=1;

                                        while(menu2<=3) {
                cout<<endl<<endl<<endl;
                cout<<"\t\t\t\t\t________M E N U________"<<endl<<endl;
        cout<<"\t\t\t\t\t01. Sign up librarians"<<endl<<endl;
        cout<<"\t\t\t\t\t02. Change the fine rate"<<endl<<endl;
    cout<<"\t\t\t\t\t03. Modify maximum borrowing limits"<<endl<<endl;
        cout<<"\t\t\t\t\t04. Show Librarian List"<<endl<<endl;
cout<<"\t\t\t\t\t05. Show fine rate and maximum borrowing limits"<<endl<<endl;
                cout<<"\t\t\t\t\t06. Exit"<<endl<<endl;
        cout<<"\t\t\t\t\t07. Back to menu"<<endl<<endl;
```

```
        cout<<"\t\t\t\t\tPleace Choose Any Option ";
    cin>>menu2;
        cout<<endl;

                                        switch(menu2) {
                                        case 1:{
cout<<endl<<endl<<"\t\t\t\t\tEnter librarian's first name:\t";
cin>>lname;
cout<<endl<<"\t\t\t\t\tEnter librarian's last six digits of NIC number:\t";
                                                cin>>lnic;
                                        if(system("CLS")) system("clear");
    length=lnic.length();
                                                if(length<6 || length>6) {

    cout<<endl<<"\t\t\t\t\tInvalid enter Try again";
                                                        menu2=1;
                                                        break;
                                                }
                                        if(system("CLS")) system("clear");
                                                signUpLibrarians(lname, lnic);
                                                        break;
                                                }
                                                case 2:{

    cout<<endl<<endl<<"\t\t\t\t\tEnter value for fine rate:\t";
                                        cin>>fineRate;
                                        if(system("CLS")) system("clear");
                                                fstream ratefile;

    ratefile.open("finerate.dat",ios::out);
                                                if(! ratefile) {

    cout<<endl<<"\t\t\t\t\tFile does not exit!";
```

```
                                                    }
                                                    else {

        ratefile<<fineRate<<"\n";
                                                                ratefile.close();

        cout<<endl<<"\t\t\t\t\tFine rate updated"<<endl;
                                                    }
                                                    break;
                                                }
                                                case 3:{
                        cout<<"\t\t\t\t\tModify maximum borrowing limits:\t";
                                                    cin>>maximumLimit;
                                            if(system("CLS")) system("clear");
                                                fstream maximumborrowfile;

        maximumborrowfile.open("maximumborrow.dat",ios::out);
                                                    if(! maximumborrowfile)
{

        cout<<endl<<"\t\t\t\t\tFile does not exit!";
                                                }
                                                    else {

        maximumborrowfile<<maximumLimit<<"\n";

        maximumborrowfile.close();

        cout<<endl<<endl<<"\t\t\t\t\tMaximum borrowing limit
updated"<<endl;
                                                    }
                                                    break;
                                                }
```

```
                                        case 4:{
                                    if(system("CLS")) system("clear");
                                                showlibrianlist();
                                                break;
                                        }

                                        case 5:{
                                    if(system("CLS")) system("clear");

        cout<<endl<<endl<<"\t\t\t\t\tFinerate And Maximum Borrowing
Limits";
        ShowFinerateAndMaximumBorrowingLimits();
                                                break;
                                        }

                                        case 6:{
                                    if(system("CLS")) system("clear");

        cout<<endl<<"\t\t\t\t\tThankyou....."<<endl;

        menu1=3;
        break;                                              }
                                        case 7:{
                                    if(system("CLS")) system("clear");
                                                break;
                                        }
                                        default:{
                                    if(system("CLS")) system("clear");

        cout<<endl<<endl<<"\t\t\t\t\tInvalid enter!";
```

```
        menu1=1;
        break;                                                  }
                                        }

                                    }
                                }
                                else {
                                    cout<<endl<<endl<<"\t\t\t\t\tWrong
password! Try again...";
                                    menu1=1;
                                }

                            }
  else if(menu1==2) {     int count=0;
cout<<endl<<endl<<"\t\t\t\t\tEnter username and password..."<<endl;
                                cout<<endl<<"\t\t\t\t\tUsername>>\t";
                                cin>>username;
                                cout<<endl<<"\t\t\t\t\tPassword>>\t";
                                cin>>password;
                                if(system("CLS")) system("clear");

                                fstream signfile;
                                signfile.open("librarian.dat",ios::in);
                    if(!signfile) {
                                    cout<<"\t\t\t\t\tFile cannot be open !!!";
                                }
                                else {
                                    /*read librarian file to get librarians
                                    user names and passwords*/
                                string line;
                                    while(getline(signfile, line)) {
                                        istringstream iss(line);
```

```
                                        getline(iss, name, ',');
                                    getline(iss, nic, ',');
                                getline(iss, fusername, ',');
                                        getline(iss, fpassword, ',');

                                        if(fpassword==password &&
fusername==username) {
                                                count=count+1;
                                        }

                                }
                        }
                                signfile.close();

                                if(count==1) {
                                int flag=1;
                            while(flag==1) {
                         int addOrDelete;
                                        cout<<endl<<endl<<endl;
                        cout<<"\t\t\t\t\t_________M E N U_________"<<endl<<endl;
                        cout<<"\t\t\t\t01. Show Book deatails"<<endl<<endl;
                        cout<<"\t\t\t\t02. Add and delete books"<<endl<<endl;
                        cout<<"\t\t\t\t03. Search and update books"<<endl<<endl;
                        cout<<"\t\t\t\t04. Issue books"<<endl<<endl;
                        cout<<"\t\t\t\t05. Accept returned books"<<endl<<endl;
                cout<<"\t\t\t\t06. Show boorrower Deatails"<<endl<<endl;
                cout<<"\t\t\t\t07. Add and delete borrowers"<<endl<<endl;
                cout<<"\t\t\t\t08. Search and update borrowers"<<endl<<endl;
                cout<<"\t\t\t\t09. Show The Prize Winner"<<endl<<endl;
                        cout<<"\t\t\t\t10. Back to menu"<<endl<<endl;
                                        cout<<"\t\t\t\tChoose Any Option: ";
                                        cin>>menu3;
                                        if(system("CLS")) system("clear");
```

```
                                addOrDelete=1;
            searchORupdate=1;
        addDeleteBorrowers=1;
        searchUpdateBorrowers=1;
                                switch(menu3) {

                                    case 1:{
                                showbooks();
                                        break;
                                    }

                                    case 2: {
                                        while(addOrDelete<=2) {

        cout<<endl<<endl<<"\t\t\t\t\t***Add And Delete Books***"<<endl;

        cout<<endl<<endl<<"\t\t\t\t\tDo you want to"<<endl;

        cout<<endl<<"\t\t\t\t\t01. Add Books"<<endl;
cout<<endl<<"\t\t\t\t\t02. Delete Books"<<endl;

        cout<<endl<<"\t\t\t\t\t03. Back To Menu"<<endl;

        cout<<endl<<"\t\t\t\t\t04. Exit"<<endl;

        cout<<endl<<"\t\t\t\t\tPleace choose any option: ";
                                                cin>>addOrDelete;
                                            if(system("CLS")) system("clear");
                                                if(addOrDelete==1) {
                                                    addbook();
                                                addOrDelete=1;
                                            }
                                                else if(addOrDelete==2) {
```

```
                                                deletebook();
                                                addOrDelete=1;
                                        }
                                else if(addOrDelete<1 || addOrDelete>4) {

        cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";
                                                addOrDelete=1;
                                        }
                                        else if(addOrDelete==4) {
                                                menu1=3;

        cout<<endl<<endl<<"\t\t\t\t\tThankyou....."<<endl<<endl<<endl<<e
ndl;
                                                flag=0;
                                        }
                                }
                                if(addOrDelete==3){
                                                flag=1;
                                        }
                                break;
        }

                                case 3:{
                                        flag=0;
                                        while(searchORupdate<=2) {
cout<<endl<<endl<<"\t\t\t\t\t***Search And Update Books***"<<endl;

        cout<<endl<<endl<<"\t\t\t\t\tDo you want to"<<endl;

        cout<<endl<<"\t\t\t\t\t01. Search Books"<<endl;

        cout<<endl<<"\t\t\t\t\t02. Update Books"<<endl;
```

```
        cout<<endl<<"\t\t\t\t\t03. Back To Menu"<<endl;

        cout<<endl<<"\t\t\t\t\t04. Exit"<<endl;

        cout<<endl<<"\t\t\t\t\tPleace choose any option: ";
                                                        cin>>searchORupdate;
                                                if(system("CLS")) system("clear");
                                                        if(searchORupdate==1) {
                                                                searchbook();

        searchORupdate=1;
                                                        }
                                                        else
if(searchORupdate==2) {
                                                                updatebook();

        searchORupdate=1;
                                                        }
                                                        else
if(searchORupdate<1 || searchORupdate>4) {
cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";

        searchORupdate=1;
                                }
                                                        else
if(searchORupdate==4) {
                                                                menu1=3;

        cout<<endl<<endl<<"\t\t\t\t\tThankyou"<<endl<<endl<<endl<<endl;
                                                                flag=0;
                                                        }
                                                }
                                                if(searchORupdate==3){
```

```
                                        flag=1;
                                    }
                                break;
                            }


                            case 4:{
                                flag=0;
                                while(issuebooks<=3) {

cout<<endl<<endl<<"\t\t\t\t\t***Issue Books***"<<endl;

cout<<endl<<endl<<"\t\t\t\t\tDo you want to"<<endl;

cout<<endl<<"\t\t\t\t\t01. Show Books"<<endl;

cout<<endl<<"\t\t\t\t\t02. Issue Books"<<endl;

cout<<endl<<"\t\t\t\t\t03. Back To Menu"<<endl;

cout<<endl<<"\t\t\t\t\t04. Exit"<<endl;
cout<<endl<<"\t\t\t\t\tPleace choose any option: ";
                                        cin>>issuebooks;
                                if(system("CLS")) system("clear");
                                        if(issuebooks==1) {
                                        showbooks();

cout<<"\t\t\t\t\tEnter No:01 To Issue Books: ";
                                            cin>>issuebooks;

                                        }
                                        if(issuebooks==2) {
                                            void issuebooks();
```

```
                                                issuebooks();

        cout<<endl<<endl<<"\t\t\t\t\tThankyou.....";
                                                }
                                        else if(issuebooks<1 || issuebooks>4) {

        cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";

        searchORupdate=1;
                                                }
                                                else if(issuebooks==4) {
                                                        menu1=3;

        cout<<endl<<endl<<"\t\t\t\t\tThankyou....."<<endl<<endl<<endl<<e
ndl;
                                                        flag=0;
                                                }
                                                if(issuebooks==3){
                                        flag=1;
                                                        break;
                                                }
                                        }
                                        break;
                                }


                                case 5:{
                                        acceptReturnedBooks();

        cout<<endl<<endl<<"\t\t\t\t\tThankyou.....";
                                        break;
                                }
```

```
                                        case 6:{
                                                showborrowerdeatails();
                                                break;
                                        }

                                        case 7:{
                                                while(addDeleteBorrowers<=2)
{
cout<<endl<<endl<<"\t\t\t\t\t***Add And Delete Borrowers***"<<endl;

        cout<<endl<<endl<<"\t\t\t\t\tDo you want to"<<endl;

        cout<<endl<<"\t\t\t\t\t01. Add Borrowers"<<endl;

        cout<<endl<<"\t\t\t\t\t02. Delete Borrowers"<<endl;

        cout<<endl<<"\t\t\t\t\t03. Back To Menu"<<endl;

        cout<<endl<<"\t\t\t\t\t04. Exit"<<endl;

        cout<<endl<<"\t\t\t\t\tPleace choose any option: ";

        cin>>addDeleteBorrowers;
                                        if(system("CLS")) system("clear");


        if(addDeleteBorrowers==1) {
                                                        addborrowers();

        addDeleteBorrowers=1;
                                }
                                                        else
if(addDeleteBorrowers==2) {
```

```
                                                    deleteBorrowers();

        addDeleteBorrowers-1;
                                                }
                                                else
if(addDeleteBorrowers<1 || addDeleteBorrowers>4) {

        cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";

        addDeleteBorrowers=1;
                                }
                                                else
if(addDeleteBorrowers==4) {
                                                        menu1=3;

        cout<<endl<<endl<<"\t\t\t\t\tThankyou....."<<endl<<endl<<endl<<e
ndl;
                                                        flag=0;
                                                }

        if(addDeleteBorrowers==3) {
                                                        flag=1;
                                                }
                                                        break;
                                                }

                                                        break;
                                                }


                                                case 8:{
```

```
        while(searchUpdateBorrowers<=2) {

        cout<<endl<<endl<<"\t\t\t\t\t***Search And Update
Borrowers***"<<endl;

        cout<<endl<<endl<<"\t\t\t\t\tDo you want to"<<endl;

        cout<<endl<<"\t\t\t\t\t01. Search Borrowers"<<endl;

        cout<<endl<<"\t\t\t\t\t02. Update Borrowers"<<endl;

        cout<<endl<<"\t\t\t\t\t03. Back To Menu"<<endl;

        cout<<endl<<"\t\t\t\t\t04. Exit"<<endl;

        cout<<endl<<"\t\t\t\t\tPleace choose any option: ";

        cin>>searchUpdateBorrowers;
        if(system("CLS")) system("clear");


        if(searchUpdateBorrowers==1) {

         searchBorrowers();

         searchUpdateBorrowers=1;
                                                        }
                                                        else
if(searchUpdateBorrowers==2) {
```

```
        UpdateBorrowers();

        searchUpdateBorrowers-1;
                                }
                                                        else
if(searchUpdateBorrowers<1 || searchUpdateBorrowers>4) {

        cout<<endl<<endl<<"\t\t\t\t\tInvalid Enter. Try Again....";

    searchUpdateBorrowers=1;
                                }
                                                        else
if(searchUpdateBorrowers==4) {
                                                                menu1=3;

 cout<<endl<<endl<<"\t\t\t\t\tThankyou....."<<endl<<endl<<endl<<e ndl;
                                                                flag=0;
                                                        }

        if(searchUpdateBorrowers==3) {
                                                                flag=1;
                                                        }
                                                            break;
                                                    }

                                                            break;
                                                }
                                                case 9:{
                                                    prizewinner();
                                                    break;
                                }

                                                case 10:{
```

```
                                                    if(system("CLS"))
    system("clear");
                            flag=0;
                            break;
                        }
                                    }
                                }
                                }

                                else {
                                        cout<<endl<<endl<<endl<<"\t\t\t\t\tInvalid
    username or password"<<endl<<endl;
                                    menu1=1;
                        }
                    }
                    else if(menu1==3) {
                            cout<<endl<<"\t\t\t\t\tThankyou....."<<endl;
                    }
                            else {
                                cout<<endl<<endl<<"\t\t\t\t\tInvalid enter!";
                            menu1=1;
                }
                }

            return 0;
    }
```