**A PROJECT REPORT**

**ON**

**Library Management System**

***submitted in partial fulfilment for the degree***

**Bachelor of Technology**

***in***

**Information Technology**

**Dr. AMBEDKAR INSTITUTE OF TECHNOLOGY FOR**

**HANDICAPPED, U.P., KANPUR**



**Under the guidance of**

**Abhishek Agnihotri Sir,**

**Head Of Department**

**SUBMITTED BY**

**HARIOM NATHANI**

**(2001660130025)**

**&**

**MANMAY GHOSH**

**(2001660130033)**

DECLARATION

We hereby declare that the project submitted by us is our own work and that to the best of our knowledge. It contains no material previously published or written by another person, nor material which to a substantial extent has been expected for the award of our other degree or diploma of the university or other institute of higher learning except where due acknowledgment has been made in the text.

**Signature:**

**Name:    Hariom Nathani**

**Roll no.: 2001660130025**

**Signature:**

**Name:    Manmay Ghosh**

**Roll no.: 2001660130033**

CERTIFICATE

This is to certify that the project report entitled ‘‘**Hospital Management System**” submitted by **Hariom Nathani and Manmay Ghosh**. In partial fulfilment for the award of ‘‘Bachelor of Technology’’ in ‘‘information technology” during the academic year 2022 has been carried out under my supervision and this work has not been submitted elsewhere for the award of any degree.

Mr. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Astt. Prof. IT Department

Dr. A.I.T.H. Kanpur

(Project Supervisor)

Date: Examiner Signature:

**ACKNOWLEDGEMENT**

We deeply thank our project guide, **Mr. Abhishek Agnihotri Sir**,for his valuable guidance. We would like to express our feelings of thankfulness to our project guide for giving his content encouragement and every facility that we required during our project. We are thankful to him for his gentle encouragement and pain taken in while guiding us through the project duration.

We are also thankful to all the faculty members of the IT Department for their encouragement and support during our project duration.

**HARIOM NATHANI**

**(2001660130025)**

**MANMAY GHOSH**

**(2001660130033)**

**TABLE OF CONTENT**

1. INTRODUCTION
2. PURPOSE
3. DETAILS ABOUT ALL THE FUNCTION USED IN THE PROJECT
4. ADVANTAGES
5. DISADVANTAGES
6. SNAPSHOT OF OUTPUT RESULT
7. CONCLUSION
8. CODE
9. POWERPOINT PRESENTATION
10. REFERENCES

# INTRODUCTION TO LIBRARY MANAGEMENT SYSTEM

* **A library management system is software that is designed to manage all the functions of a library.**
* **It helps librarian to maintain the database of new books and the books that are borrowed by members along with their due dates.**
* **This system completely automates all your library's activities**

**Abstract**

Our library management is meant to help librarian to maintain a record of books, issue books to students and keep a track of students and the books issued too it so fine can be count on.

It can only be accessed by the concerned authorities so that no else can manipulate with its data.

PURPOSE

The purpose of a library management system is to operate a library with efficiency and at reduced costs. The system being entirely automated streamlines all the tasks involved in operations of the library. The activities of book purchasing, cataloging, indexing, circulation recording and stock checking are done by the software. Such software eliminates the need for repetitive manual work and minimizes the chances of errors.  
  
The library management system software helps in reducing operational costs. Managing a library manually is labor intensive and an immense amount of paperwork is involved. An automated system reduces the need for manpower and stationery. This leads to lower operational costs.  
  
The system saves time for both the user and the librarian. With just a click the user can search for the books available in the library. The librarian can answer queries with ease regarding the availability of books. Adding, removing or editing the database is a simple process. Adding new members or cancelling existing memberships can be done with ease.

DETAILS ABOUT ALL THE FUNCTION USED IN THE PROJECT

1. **Use of add\_books and view\_books function**

* add\_books function opens the binary file in append mode and writes the book and its details. Once you enter a detail it stores in the file but did not delete the previous record.
* view\_books function opens the file in reading mode and read and display all the stored book details. If there is no book available in the records, then it displays the message record is empty.

1. **Use of Search Books case**

* Search Books case opens the binary file in reading mode and asks the user to enter either the book id or book name which user wants to search.
* If the book is available in the list, it will show the details of the book.

1. **Use of issue\_books function**

* issue\_books function opens the binary file in append mode and writes the book and student details.
* Once you enter a detail it stores in the file but did not delete the previous record.
* Once the book is issued it tells that when the books is to be return and rent for the book to be issued.

1. **Use of update\_books function**

* update\_books function opens the binary file in read mode and display the book name and student details.
* If you choose to reissue the book it will also display return date and rent to be paid for next issue(reissue).
* If you choose to submit the book it will clear all the details from the file so tha next some one else can issue it.

1. **Use of students\_details function**

* students\_details function opens the binary file in read mode and display the book and student details.
* Since it is open in read mode no change in detail is allowed in this function keeping the rest record as same as it was earlier without any change and display specific results.

1. **Use of change password function**

* change\_password function opens the file in rb+ mode (reading and writing). It asks the user for the new password.
* After taking the password it stores the password in another file to make it more secure.
* Now user can use the application with a new password and original username.

**ADVANTAGES**

1. Simple & Easy to Use
2. Increased Library Engagement
3. Dynamic Reports
4. Highly Secure, Scalable & Reliable
5. Innovation
6. Fully Customizable
7. Cost-effective

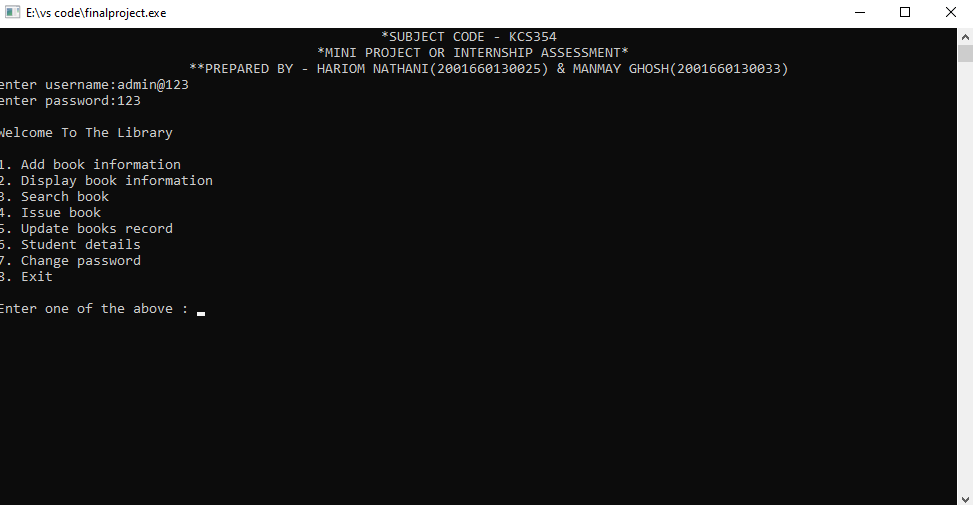
**Disadvantage**

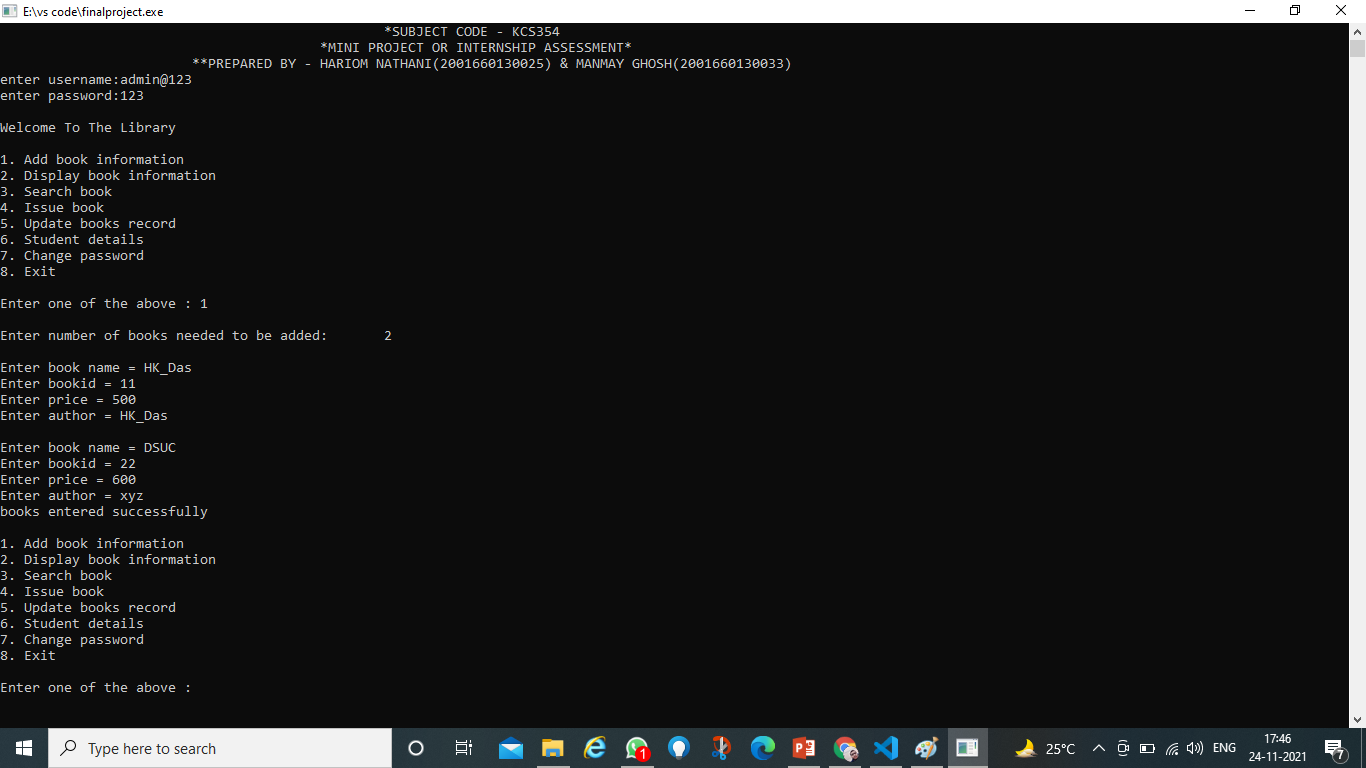
1. The data stored is prone to cyber hacks.
2. Complicated to operate
3. Risk of computer virus
4. Requires manual action to perform operations
5. No remote accessibility
6. Risk of manipulation

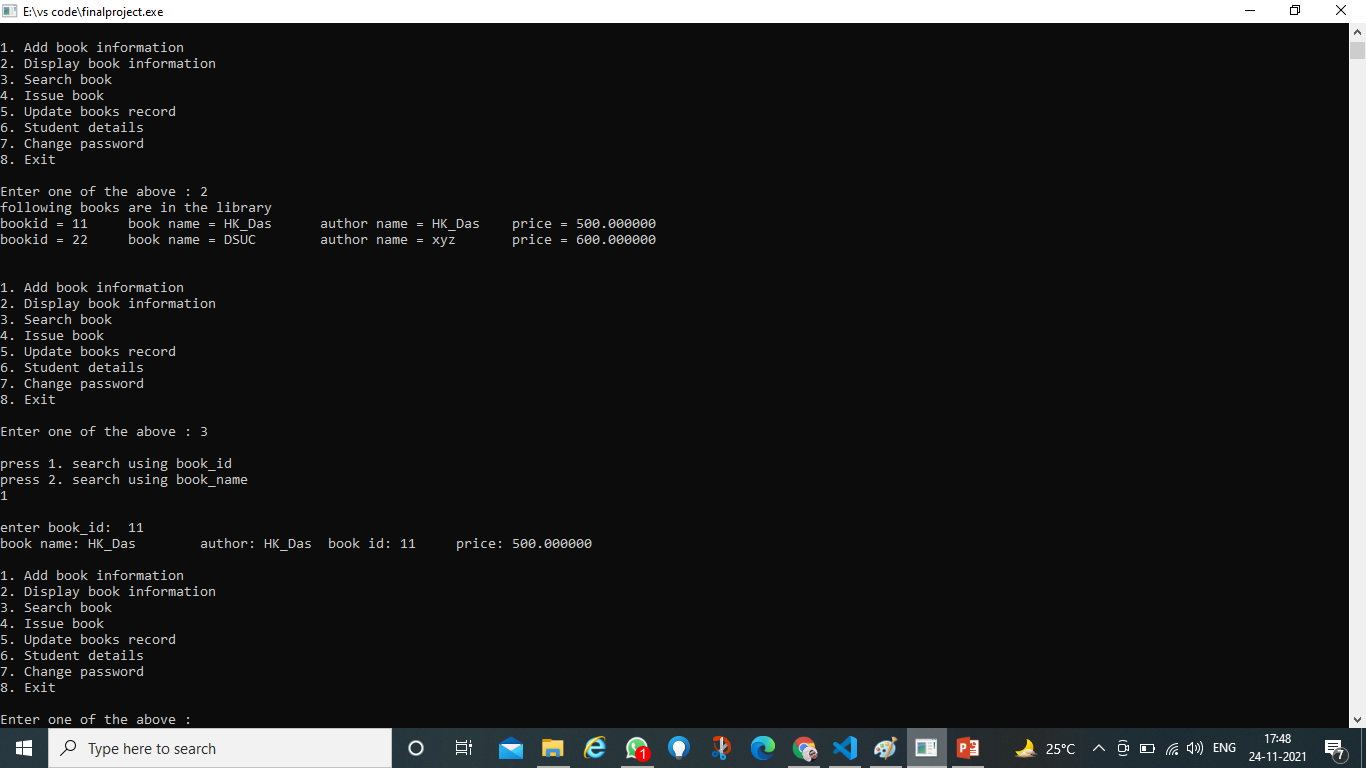
**SOME SNAPSHOT**

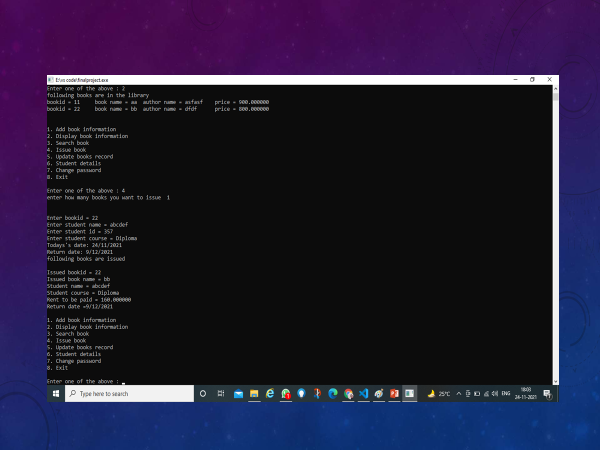
**OF**

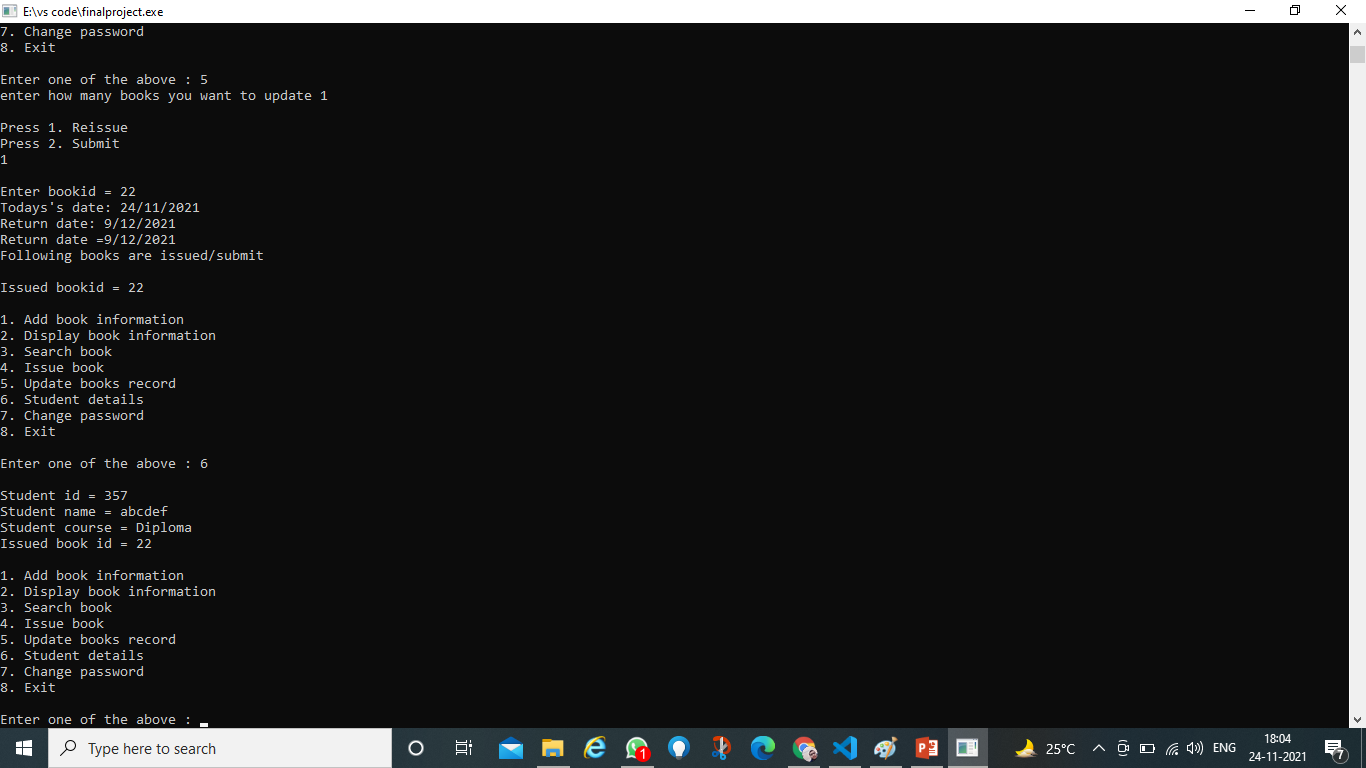
**OUTPUT RESULT**











CONCLUSION

**The Library Management System allows the user to store the book details and the person's details.**

**This software allows storing the details of all the data related to library.**

**The implementation of the system will reduce data entry time and provide readily calculated reports.**

References

1. Basic designing of a library management system

https://www.sciencedirect.com/topics/computer-science/library-management-system/

1. Purpose of a library management system

<https://www.skoolbeep.com/blog/library-management-system/>

1. Some other library management system

* <https://www.educative.io/courses/grokking-the-object-oriented-design-interview/RMlM3NgjAyR>
* <https://www.daitm.org.in/wp-content/uploads/2019/04/Gr.-06library-project-report.pdf>
* <https://en.wikipedia.org/wiki/Library_management>
* <https://www.edureka.co/blog/library-management-system-project-in-java>
* <https://www.academia.edu/37726542/Library_Management_System_Mini_Project_Report_On_LIBRARY_MANAGEMENT_SYSTEM>
* <https://www.slideshare.net/smitpatel10192/library-management-system-presentation>
* <https://github.com/topics/library-management-system>
* https://www.geeksforgeeks.org/e-library-management-system/

Code for library management system

#include<stdio.h>

#include<string.h>

#include<conio.h>

#include<stdlib.h>

#include<windows.h>

void add\_books(int count);

void book\_record();

void issue\_books();

void update\_record();

void student\_details();

void change\_password();

int temp\_bookId;

int temp;

struct library

{

int book\_id;

char book\_name[30];

char author[30];

float price;

char issued\_status[10];

}om[100];

struct date

{

int da\_day;

int da\_mon;

int da\_year;

}submit;

struct students

{

char student\_name[20];

int student\_id[20];

char student\_course[20];

int issued\_bookid[20];

struct date submit\_date;

}std[100];

int main()

{

int j,i=0,count,n=0,m;

char username[20];

char password[20];

char originalpassword[20];

printf("\n");

printf("\t\t\t\t\t\*MINI PROJECT OR INTERNSHIP ASSESSMENT\*\n");

printf("\t\t\t\*\*PREPARED BY - HARIOM NATHANIIT DEPT, DR.AITH, KANPUR & MANMAY GHOSH, IT DEPT, DR.AITH, KANPUR\n");

//authentication

printf("enter username:");

scanf("%s", &username);

printf("enter password:");

scanf("%s",&password);

FILE \*fp;

fp=fopen("password.txt","r");

fgets(originalpassword,10,fp);

fclose(fp);

if((strcmp(username,"admin@123")==0)&&(strcmp(password,originalpassword)==0))

{

printf("\nWelcome To The Library");

while(j!=9)

{

printf("\n\n1. Add book information\n");

printf("2. Display book information\n");

printf("3. Search book\n");

printf("4. Issue book\n");

printf("5. Update books record\n");

printf("6. Student details\n");

printf("7. Change password\n");

printf("8. Exit\n");

printf ("\n\n\nEnter one of the above : ");

scanf("%d",&j);

switch (j)

{

case 1: //printf("\nEnter number of books needed to be added:\t");

// scanf("%d",&count);

count=1;

add\_books(count);

break;

case 2: book\_record(count);

break;

case 3: printf("press 1. search using book\_name");

printf("\npress 2. search using book\_id\n");

scanf("%d",&n);

fp = fopen("booklib.txt","rb");

if(fp == NULL)

{

printf("\n\t\t\tFile is not opened\n");

exit(1);

printf("%d", (fread (&om, sizeof(om), 1, fp)));

}

if(n==1)

{

char book\_name[30];

printf ("Enter book name:\t ");

scanf ("%s",&book\_name);

// for (i=0; i<count; i++)

while (fread (&om, sizeof(om), 1, fp))

{

// printf(" %d", fread (&om, sizeof(om), 1, fp));

//printf("%d ", om[i].book\_id);

if (strcmp(book\_name, om[i].book\_name) == 0)

printf ("book name: %s\t author: %s\t book id: %d\t price: %f",om[i].book\_name,om[i].author,om[i].book\_id,om[i].price);

}

printf("%d",fread (&om, sizeof(om), 1, fp));

}

else if(n==2)

{

int book\_id1;

printf("\nenter book\_id:\t");

scanf("%d",&book\_id1);

while (fread (&om, sizeof(om), 1, fp))

{

if (book\_id1==om[i].book\_id)

{

printf ("book name: %s\t author: %s\t book id: %d\t price: %f",om[i].book\_name,om[i].author,om[i].book\_id,om[i].price);

}

}

}

break;

case 4: issue\_books();

break;

case 5:

update\_record();

break;

case 6: student\_details();

break;

case 7: change\_password();

break;

case 8: printf("Thank You for using our library management sytem\nPress any key to exit...");

getch();

exit(0);

}

}

return 0;

}

else

printf("wrong password");

}

void add\_books(int count) //add book information

{

int i;

FILE \*fp;

fp=fopen("booklib.txt","ab+");

if(fp == NULL)

{

printf("File is not opened\n");

exit(1);

}

for(i=0; i<count; i++)

{

printf("\nEnter book name = ");

scanf("%s",om[i].book\_name);

printf("Enter bookid = ");

scanf ("%d",&om[i].book\_id);

printf("Enter price = ");

scanf("%f",&om[i].price);

printf("Enter author = ");

scanf("%s",&om[i].author);

fwrite(&om,sizeof(om),1,fp);

fclose(fp);

}

book\_record();

}

void book\_record() //display book information

{

int i=0;

FILE \*fp;

fp=fopen("booklib.txt","rb");

if(fp==NULL)

{

printf("file not found");

exit(1);

}

printf("following books are in the library\n");

while (fread (&om, sizeof(om), 1, fp))

{

// fread(&om,sizeof(om[0]),1,fp;

printf("\nbookid = %d",om[i].book\_id);

printf("\tbook name = %s",om[i].book\_name);

fflush(stdin);

printf("\tauthor name = %s",om[i].author);

printf("\tprice = %f",om[i].price);

}

fclose(fp);

}

void issue\_books()

{

int i=0,j=0;

FILE \*fp;

fp=fopen("booklib.txt","ab+");

if(fp == NULL)

{

printf("File is not opened\n");

exit(1);

}

printf("\nEnter bookid = ");

scanf ("%d",&std[i].issued\_bookid);

temp\_bookId=\*std[i].issued\_bookid;

while(fread (&om, sizeof(om), 1, fp))

{

if (\*std[i].issued\_bookid==om[j].book\_id && om[j].issued\_status=="Issued")

printf("\nThis book is not available in library");

else

{

printf("Enter student name = ");

scanf("%s",&std[i].student\_name);

printf("Enter student id = ");

scanf ("%d",&std[i].student\_id);

printf("Enter student course = ");

scanf("%s",&std[i].student\_course);

if(\*std[i].issued\_bookid==om[j].book\_id)

strcpy(om[j].issued\_status,"Issued");

tested();

std[i].submit\_date=submit;

fwrite(&std,sizeof(std),1,fp);

fwrite(&om,sizeof(om),1,fp);

fclose(fp);

}

}

printf("\n\n\nfollowing books are issued\n");

printf("\nIssued bookid = %d",temp\_bookId);

fp=fopen("booklib.txt","rb");

while(fread (&om, sizeof(om), 1, fp))

{

if (temp\_bookId==om[i].book\_id)

{

printf ("\nbook name: %s\nauthor: %s\nbook id: %d\nprice: %f",om[i].book\_name,om[i].author,om[i].book\_id,om[i].price);

fread (&std, sizeof(std), 1, fp);

{

// printf("\nIssued book name = %s",om[j].book\_name);

printf("\nStudent name = %s",std[i].student\_name);

printf("\nStudent course = %s",std[i].student\_course);

printf("\nRent to be paid = %f",0.2\*om[j].price);

printf("\nReturn date =%d/%d/%d",std[i].submit\_date.da\_day,std[i].submit\_date.da\_mon,std[i].submit\_date.da\_year);

}

}

fclose(fp);

}

}

void update\_record()

{

int i=0,j,k,m,issuebook,submitdate;

FILE \*fp;

fp=fopen("booklib.txt","ab+");

if(fp == NULL)

{

printf("File is not opened\n"); ////fwrite use

exit(1);

}

printf("\nPress 1. Reissue\n");

printf("Press 2. Submit\n");

scanf("%d",&k);

printf("\nEnter bookid = ");

scanf ("%d",&issuebook);

temp= issuebook;

if(k==1)

{

while (fread (&std, sizeof(std), 1, fp))

{

if(\*std[i].issued\_bookid==issuebook)

{

tested();

printf("\nReturn date =%d/%d/%d",std[j].submit\_date.da\_day,std[j].submit\_date.da\_mon,std[j].submit\_date.da\_year);

}

}

}

else if(k==2)

{

while (fread (&std, sizeof(std), 1, fp))

{

if(\*std[i].issued\_bookid==issuebook)

{

// temp= issuebook;

for(m=0;i<strlen(std[j].student\_id);i++)

{

std[j].student\_id[m] = 0;

}

for(m=0;i<strlen(std[j].student\_name);i++)

{

std[j].student\_name[m] = 0;

}

for(m=0;i<strlen(std[j].student\_course);i++)

{

std[j].student\_course[m] = 0;

}

}

}

}

else

printf("Please enter a valid number");

fwrite(&std,sizeof(std),1,fp);

fclose(fp);

printf("\nFollowing books are reissued/submit\n");

fp=fopen("booklib.txt","rb");

printf("\nIssued bookid = %d",temp);

while(fread (&om, sizeof(om), 1, fp))

{

if (temp==om[i].book\_id)

{

printf ("\nbook name: %s\nauthor: %s\nbook id: %d\nprice: %f",om[i].book\_name,om[i].author,om[i].book\_id,om[i].price);

fread (&std, sizeof(std), 1, fp);

{

// printf("\nIssued book name = %s",om[j].book\_name);

printf("\nStudent name = %s",std[i].student\_name);

printf("\nStudent course = %s",std[i].student\_course);

printf("\nRent to be paid = %f",0.05\*om[j].price);

printf("\nReturn date =%d/%d/%d",std[i].submit\_date.da\_day,std[i].submit\_date.da\_mon,std[i].submit\_date.da\_year);

}

}

fclose(fp);

}

}

void student\_details()

{

int i=0;

FILE \*fp;

fp=fopen("booklib.txt","rb");

if(fp==NULL)

{

printf("file not found");

exit(1);

}

printf("following books are issued \n");

while (fread (&std, sizeof(std), 1, fp))

{

printf("\nStudent id = %d",\*std[i].student\_id);

printf("\nStudent name = %s",std[i].student\_name);

printf("\nStudent course = %s",std[i].student\_course);

printf("\nIssued book id = %d\n",\*std[i].issued\_bookid);

}

fclose(fp);

}

void change\_password()

{

char pwd[50];

FILE \*fp;

fp= fopen("password.txt","w");

printf("Enter New Password: ");

fseek(stdin,0,SEEK\_END);

scanf("%s",&pwd);

fputs(pwd,fp);

fclose(fp);

}

void tested()

{

int j;

SYSTEMTIME stime;

GetSystemTime(&stime);

printf("Todays's date: %d/%d/%d",stime.wDay,stime.wMonth,stime.wYear);

int noOfdayinmonth[]={0,31,28,31,30,31,30,31,31,30,31,30,31};

submit.da\_day = (int)stime.wDay+15;

submit.da\_mon = (int)stime.wMonth;

submit.da\_year = (int)stime.wYear;

if(noOfdayinmonth[(int)stime.wMonth] < submit.da\_day)

{

submit.da\_day -= noOfdayinmonth[(int)stime.wMonth];

submit.da\_mon +=1;

if(submit.da\_mon>12){

submit.da\_mon -= 12;

submit.da\_year += 1;

}

else

submit.da\_year=(int)stime.wYear;

}

else{

submit.da\_mon=(int)stime.wMonth;

submit.da\_year=(int)stime.wYear;

}

// printf("\nReturn date: %d/%d/%d",submit.da\_day, submit.da\_mon, submit.da\_year);

}