**LIBRARY MANAGEMENT SYSTEM**

**+**

**STUDENT RECORD MANAGEMENT**

**PROJECT REPORT**

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF

**BACHELOR OF TECHNOLOGY**

Computer Science And Engineering

SUBMITTED BY

(Aditya kumar tiwari-2103490100005)

September-2022



**CERTIFICATE**

This is to certify that the Project Report on the topic of "**LIBRARY MANAGEMENT + STUDENT MANAGEMENT SYSTEM**" is submitted by **Aditya Kumar Tiwari CSE 2nd year (2103490100005)** in fulfillment for the award of degree of **BACHELOR OF TECHNOLOGY** in **Computer Science and Engineering** has been found satisfactory and is approved for submission.

1. Embodies the work of the candidates themselves,
2. Has duly been completed, and
3. Is upto the desired standard both in respect of contents and language for being referred to the examiners.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mr. Subhash Maurya** **Dr. Saurabh Singh**

Associate Professor (CSE) Department Head of Department CSE

MPEC MPEC

**ACKNOWLEDGEMENT**

It gives us the great sense of pleasure to present the report of the B. Tech project undertaken during B. Tech 2nd year. I owe special debt of gratitude to **Dr. Saurabh Singh** (Head of Department in Computer Science and Engineering).

The satisfaction that accompanies the successful completion of any task would be incomplete without mention of people whose ceaseless cooperation made in possible, whose constant guidance and encouragement crown all effort with success.

We are grateful to our project guide **Mr.Subhash Maurya** for the guidance, inspiration and constructive suggestions that help us in the preparation in this project.

I also thank my college and colleagues who helped us in successful completion of the project.

**ADITYA KUMAR TIWARI**

**(2103490100005)**

**Dr. APJ Abdul Kalam Technical University, Uttar Pradesh**

LUCKNOW, INDIA

**ABSTRACT**

This is a “Library Management System + Student record management” project which is created using C++ programming language. We can use this simple mini project to boost our programming skills and understand logic. The Program allows the data to be recorded very easily and can be seen at a later time. The code can be accessed by the faculty who wishes to put the student’s record by putting the correct username and password . We can also see the information of all the books or student data at a same time. The record of any individual book or student can be seen by just choosing the option and then typing their name. We can also make changes in someone’s record at any time we want. It is also possible to delete anyone’s by just typing the name of an individual. The code is very user friendly and simple to use.

**TABLE OF CONTENT**

[INTRODUCTION 6](#_Toc120744732)

[SYSTEM REQUIREMENT 8](#_Toc120744733)

[1.1 Software Requirement 8](#_Toc120744734)

[1.2 Hardware Requirement 8](#_Toc120744735)

[IMPLEMENTATION AND RESULTS 9](#_Toc120744736)

[2.1 Flowchart 9](#_Toc120744737)

[Technologies Used 11](#_Toc120744738)

[3.1 What is C++? 11](#_Toc120744739)

[3.2 Why Use C++ 11](#_Toc120744740)

[3.2 Difference between C and C++ 12](#_Toc120744741)

[3.4 C++ Header Files 12](#_Toc120744742)

[3.5 FEATURES OF C++ 12](#_Toc120744743)

[3.6 Application of c++ 13](#_Toc120744744)

[Result 15](#_Toc120744745)

[4.1 Project Link 20](#_Toc120744746)

[CONCLUSION 21](#_Toc120744747)

[5.1 Result analysis 22](#_Toc120744748)

[Future Scope 23](#_Toc120744749)

[Reference 24](#_Toc120744750)

[Appendix Code 26](#_Toc120744751)

# INTRODUCTION

The mini-project “Library management system + Student record management project in C++” is a console application using the C++ programming language. This project compiled in Code Blocks with the GCC compiler. In this console application, you can do basic library management task as well as basic student management task like adding the book, view the added book, search the books,add student data,delete stuent data, ..etc.

This application based on file handling in C, where I have used a file-related function like **[fopen](https://aticleworld.com/fopen-in-c/)**, **[fread](https://aticleworld.com/fread-in-c/)**, **[fwrite](https://aticleworld.com/fwrite-in-c/)**, ..etc. Good thing is that “Library management system project” is password-protected, so only authorized person able to login in this application.

Also to increase the redabilty I have broken the application in different function. Each function of the project extensively use the [**file handing function**](https://aticleworld.com/file-handling-in-c/), so it is also a great project to understand file handling

**OBJECTIVE**

**AIM:** To make a Program with the help of C++ Programming language which can perform different types of Operations like Adding the book, Adding the record of student update the book or student record, deleting any particular record.

By choosing different options on output screen , you can make perform various operations continuously, even the user can work again and again until he want to exit.

If he wants to exit he can perform the exit operation given on the output screen.

# SYSTEM REQUIREMENT

## 1.1 Software Requirement

* Dev c++ - Version 4.9.2 - Release 2021-05-19
* Turbo c++ -Version-3.7.8.9

## 1.2 Hardware Requirement

* **Operating System:**

Window 7 and later

* **Processor:**

Intel dual core

* **Disk Space:**

1GB or more

# IMPLEMENTATION AND RESULTS

## 2.1 Flowchart

**ADMIN**

**LOGIN**

**USERNAME**

**PASSWORD CORRECT**

**LIBRARY MANAGEMENT SYSTEM OR STUDENT RECORD MANAGEMENT SYSTEM**

**1.ADD A BOOK**

**2. SEARCH ALL BOOK**

**3.DISPLAY ALL BOOKS**

**4.DELETE A BOOK**

**5.UPDATE A BOOK**

**1.ADD A STUDENT**

**2.SEARCH A STUDENT**

**3.DISPLAY ALL RECORDS**

**4.DELETE A RECORD**

**5.UPDATE A RECORD**

**6.QUIT**

**6.QUIT**

# Technologies Used

## What is C++?

C++ is a cross-platform language that can be used to create high-performance applications.

C++ was developed by Bjarne Stroustrup, as an extension to the [C language](https://www.w3schools.com/c/index.php).

C++ gives programmers a high level of control over system resources and memory.

The language was updated 4 major times in 2011, 2014, 2017, and 2020 to C++11, C++14, C++17, C++20.

## 3.2 Why Use C++

C++ is one of the world's most popular programming languages.

C++ can be found in today's operating systems, Graphical User Interfaces, and embedded systems.

C++ is an object-oriented programming language which gives a clear structure to programs and allows code to be reused, lowering development costs.

C++ is portable and can be used to develop applications that can be adapted to multiple platforms.

C++ is fun and easy to learn!

As C++ is close to [C#](https://www.w3schools.com/cs/index.php) and [Java](https://www.w3schools.com/java/default.asp), it makes it easy for programmers to switch to C++ or vice versa.

## Difference between C and C++

C++ was developed as an extension of [C](https://www.w3schools.com/c/index.php), and both languages have almost the same syntax.

The main difference between C and C++ is that C++ support classes and objects, while C does not.

## C++ Header Files

There are 2 types of header files such as:-

**Standard Library Header Files:-** These are pre-existing header files which are available in C.

**User-Defined Header Files:-** The header files which are defined by the user then they are called user-defined header files. The #define directive is used to define a header file.

## 3.5 FEATURES OF C++

C++ is an upgraded version of [C programming](https://www.edureka.co/blog/c-programming-tutorial/). The main idea behind creating [C++ programming](https://www.edureka.co/blog/cpp-tutorial/) was to add object orientation to the C programming language. The major upgradations are [object-oriented programming](https://www.edureka.co/blog/object-oriented-programming-in-cpp/) methodology, namespace feature, operator overloading, error & exception handling. The motivation behind object-oriented programming is to try to see the whole world in the form of classes & objects.

There are various features of C++ such as,

* [Object Oriented](https://www.edureka.co/blog/features-of-cpp/)
* [Simple](https://www.edureka.co/blog/features-of-cpp/)
* [Platform Dependent](https://www.edureka.co/blog/features-of-cpp/)
* [Mid-level programming language](https://www.edureka.co/blog/features-of-cpp/)
* [Structured programming language](https://www.edureka.co/blog/features-of-cpp/)
* [Rich Library](https://www.edureka.co/blog/features-of-cpp/)
* [Memory Management](https://www.edureka.co/blog/features-of-cpp/)
* [Powerful & Fast](https://www.edureka.co/blog/features-of-cpp/)
* [Pointers](https://www.edureka.co/blog/features-of-cpp/)
* [Compiler based](https://www.edureka.co/blog/features-of-cpp/)

Syntax based language

## Application of c++

C++ is a widely used programming language that is used for writing large-scale commercial applications for end-users. Some of the major applications built using C++ by major software vendors and giants are −

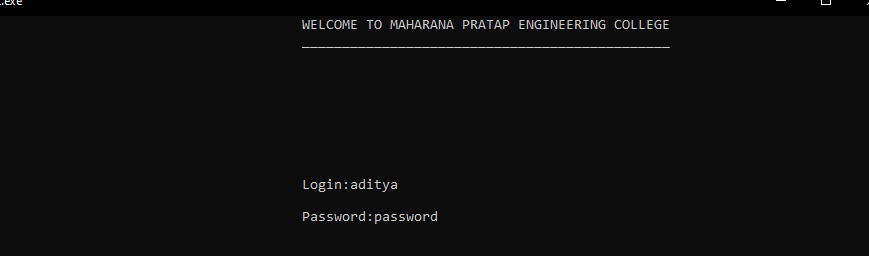
* **Google** − Google file system, Google Chromium browser, and Map Reduce large cluster data processing are all written in C++.
* **Mozilla** − Mozilla Firefox and Thunderbird email chat client are both written using C++.
* **MySQL** − MySQL, an open source DBMS is written using C++.
* **Microsoft** − Many windows apps that you regularly use are written in C++.
* **Rockstar Games** − Almost all major game companies use C++ due to its sheer speed on bare metal. Many major game engines are fully written in C++ and leverage its speed and OOP capabilities.
* **MongoDB** − An open-source database, widely used as the back end store for web applications, as well as in large enterprises like Viacom and Disney.
* **Morgan Stanley** − They use it for a huge part of their financial modeling. The creator of C++, Bjarne Stroustrup works here.

Looksmart is predominantly written in C++. All products related to searching and exploring the web are written in C++. Used by well over 5,000,000 unique users per day

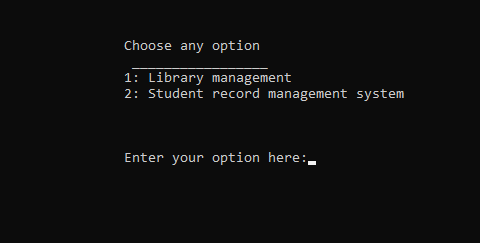
# Result

Screenshots taken from successful run of project.

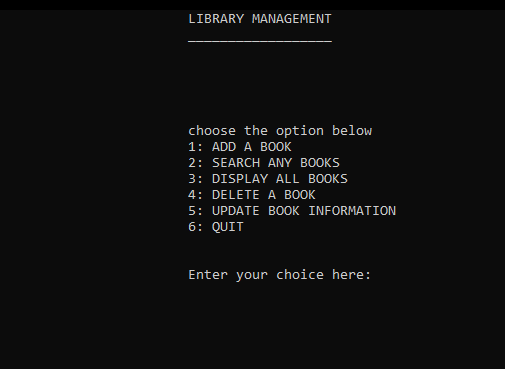
**Snapshot 1**



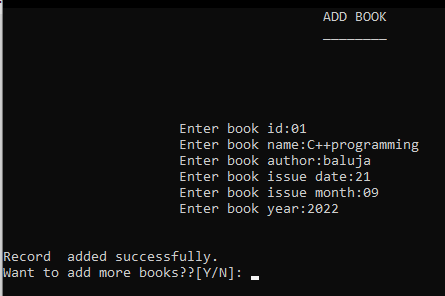
**Snapshot 2**



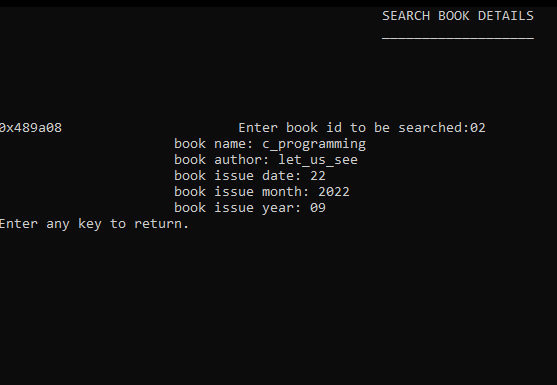
**Snapshot 3**



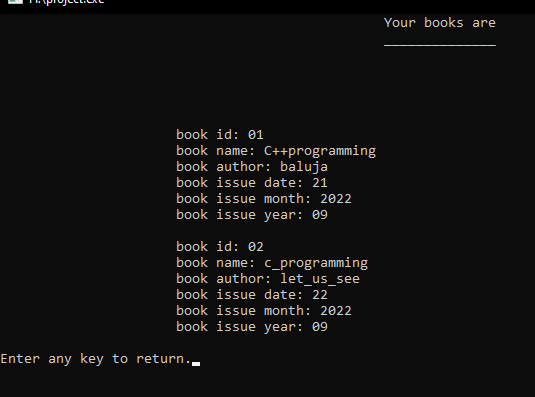
**Snapshot 4**



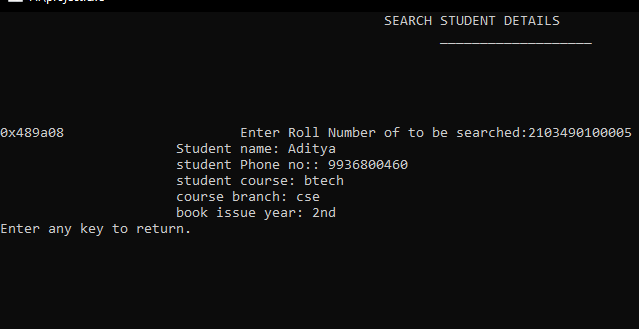
**Snapshot 5**



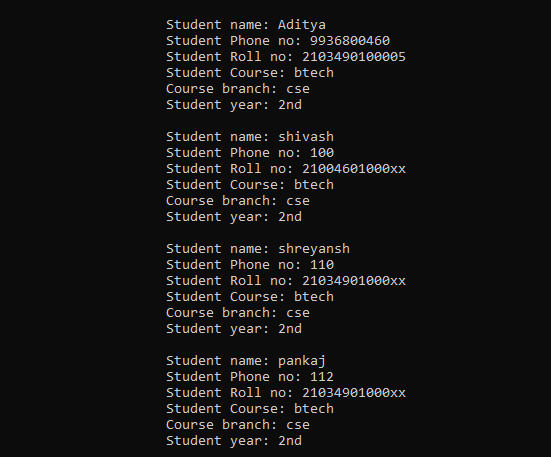
**Snapshot 6**



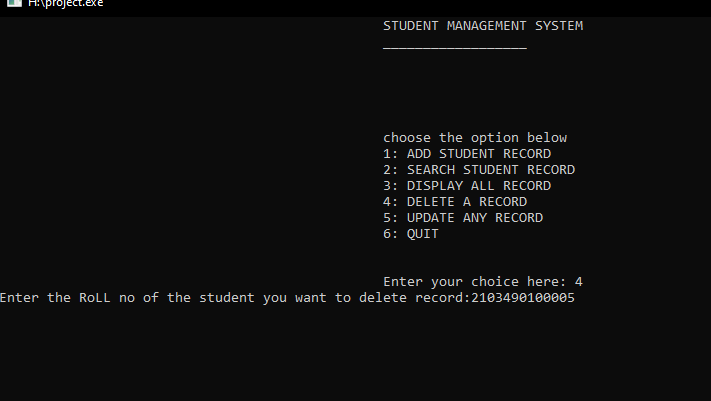
**Snapshot 7**



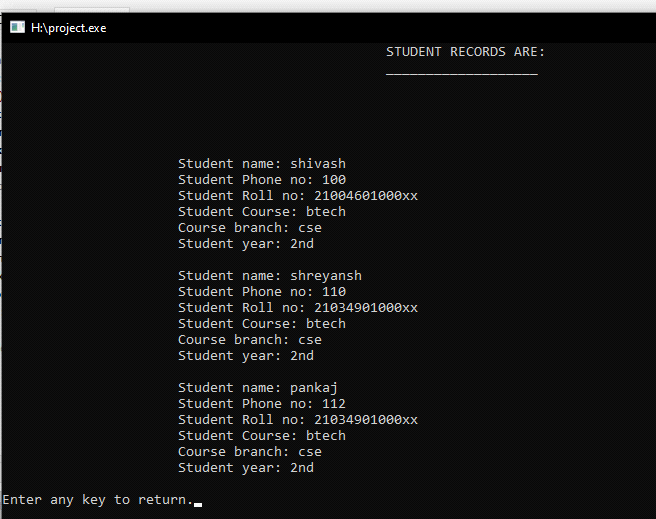
**Snapshot 8**



**Snapshot 9**



**Snapshot 10**



## 4.1 Project Link

<https://github.com/aditya8182/Project_libraray-student_manage.git>



# CONCLUSION

The Library and Student record management system is programmed with the help of C++ programming language. In this particular program you can perform different kind of operations like:-

* **Library management system**
* Add a book
* Search any book
* Display all book
* Delete a book
* Update a book
* EXIT
* **Student record management system**
* Add Student record
* Search Student record
* Display all record
* Delete a record
* Update any record
* EXIT

**NOTE:-** All the type of output need different types of input and providing the input The desired output is generated.

## 5.1 Result analysis

All the Operations were done by This project as shown in output Screenshots are correct and successfully reviewed . The Operation which are done are as follows

* Data entry of books
* Data entry of students
* Manipulation and updating the data
* Deleting the data

# Future Scope

The scope of this project is as follows:

To assist the staff in capturing the effort spent on their respective working areas. To utilize resources in an efficient manner by increasing their productivity through automation. The system generates types of information that can be used for various purposes.

# Reference

**Website:**

1. [**www.google.com**](http://www.google.com)
2. [**www.geeksforgeeks.org**](http://www.geeksforgeeks.org)
3. [**www.github.com**](http://www.github.com)
4. [**www.youtube.com**](http://www.youtube.com)
5. [**www.javatpoint.com**](http://www.javatpoint.com)

**Books:**

* Object Oriented Programming Using C++ - **G.S. BALUJA**
* Let us C++ - **Yashavant Kanetkar**
* Programming n C – **Ajay Mittal**

**Reference Paper:-**

1].HonghaiKan,Zhimin Yang, Yue Wang, Nana Qi, “Research on Library

Management System for CDs Attached to Books Based on Cloud Computing”, in

Proceedings of the 14th International Conference on Computer Supported

Cooperative Work in Design 2010.

[2].Bao Sun, JiangweiFeng and Ling Liu, “A Study on How to Construct the

Prediction Model of Library Lending of University Library”, International

Conference on Information Science and Technology March 26-28, 2011 Nanjing,

Jiangsu, China.

[3].Erxiang Chen,Minghui Liu,“Research and Design on Library Management System

Based on Struts and Hibernate Framework”, WASE International Conference on

Information Engineering2009.

[4].JianhuZheng, YunqingFeng, Yun Zhao, “A Unified Modeling Language-Based

Design and Application for a Library Management Information System”, in

cybernetics and information technologies.

[5].Michael Hitchens, Andrew Firmage,“The Design of a Flexible Class Library

Management System”, in IEEE conference 1998.

[6].WeihongYang,“Design and Implementation of Library Management System”,

International Conference on Management Science and Innovative Education

(MSIE 2015).

[7].Bretthauer, D. “Open source software in libraries. Library Hi Tech News, 18 (5),

8-9(2001).

[8].Barve, S., &Dahibhate, N. B.,“Open source software for library services”,

DESIDOC Journal of Library & Information Technology, 32(5)(2012).

# Appendix Code

#include<iostream>

#include<stdio.h>

#include<string.h>

#include <unistd.h>

#include<stdlib.h>

#include "book.cpp"

#include<string.h>

#include<conio.h>

#include "student.cpp"

using namespace std;

class book books[200];

class student s[200];

void searchbook();

void addbook();

int displaybook();

void library();

void studentmanage();

void deletebook();

void updatebook();

void addrecordrecord();

void searchrecord();

void displayrecord();

void deleterecord();

void updaterecord();

int n=0;

void addrecord()

{

FILE \*fp;

fp=fopen("studentdata.dat","w");

if(fp==NULL)

{

printf("File can'nt be open");

exit(1);

}

int i=0;

int n=0;

char ch;

char id[5],year[4];

while(1)

{

system("CLS");

cout<<"\t\t\t\tADD STUDENT RECORD"<<endl;

cout<<"\t\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\t\tEnter student name:";

cin>>s[i].name;

cout<<"\t\t\t\tEnter student Phone no:";

cin>>s[i].phoneno;

cout<<"\t\t\t\tEnter student Roll no:";

cin>>s[i].rollno;

cout<<"\t\t\t\tEnter student course:";

cin>>s[i].course;

cout<<"\t\t\t\tEnter course branch:";

cin>>s[i].branch;

cout<<"\t\t\t\tEnter student year:";

cin>>s[i].year;

fwrite(&s[i],sizeof(struct student),1,fp);

i++;

n++;

cout<<endl<<endl;

printf("Record added successfully.");

cout<<endl<<"Want to add more books??[Y/N]: ";

cin>>ch;

if(ch=='N' || ch=='n')

{

system("CLS");

break;

}

else

{

cout<<"Enter given characters only:";

}

}

fclose(fp);

return;

}

void searchrecord()

{

system("CLS");

FILE \*fp;

fp=fopen("studentdata.dat","r");

if(fp==NULL)

{

printf("File can'nt be open");

exit(1);

}

char rollid[15];

int i=0;

cout<<"\t\t\t\tSEARCH STUDENT DETAILS"<<endl;

cout<<" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<cout<<" Enter Roll Number of to be searched:";

cin>>rollid;

while(fread(&s[i],sizeof(struct student),1,fp))

{

if(!strcmp(s[i].rollno,rollid))

{

cout<<" Student name: "<<s[i].name<<endl;

cout<<" student Phone no:: "<<s[i].phoneno<<endl;

cout<<" student course: "<<s[i].course<<endl;

cout<<" course branch: "<<s[i].branch<<endl;

cout<<" book issue year: "<<s[i].year<<endl;

}

i++;

}

cout<<"Enter any key to return.";

getch();

}

void displayrecord()

{

system("CLS");

int i=0;

FILE \*fp;

fp=fopen("studentdata.dat","r");

if(fp==NULL)

{

printf("File can'nt be open");

exit(1);

}

cout<<"\t\t\t\tSTUDENT RECORDS ARE:"<<endl;

cout<<"\t\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

while(fread(&s[i],sizeof(struct student),1,fp))

{

cout<<" Student name: "<<s[i].name<<endl;

cout<<" Student Phone no: "<<s[i].phoneno<<endl;

cout<<" Student Roll no: "<<s[i].rollno<<endl;

cout<<" Student Course: "<<s[i].course<<endl;

cout<<" Course branch: "<<s[i].branch<<endl;

cout<<" Student year: "<<s[i].year<<endl<<endl;

i++;

}

fclose(fp);

cout<<"Enter any key to return.";

getch();

}

void deleterecord()

{

FILE \*fp1,\*fp2;

fp1=fopen("studentdata.dat","r+");

fp2=fopen("copy2.dat","a+");

if(fp1==NULL)

{

printf("Error in opening a file.");

}

char ids[20];

printf("Enter the RoLL no of the student you want to delete record:");

cin>>ids;

int i=0;

while(fread(&s[i],sizeof(struct student),1,fp1)){

if(strcmp(s[i].rollno,ids))

{

fwrite(&s[i],sizeof(struct student),1,fp2);

}

i++;

}

fclose(fp1);

fclose(fp2);

remove("studentdata.dat");

rename("copy2.dat","studentdata.dat");

printf("Record updated successfully.");

displayrecord();

}

void updaterecord()

{

FILE \*fp1,\*fp2;

fp1=fopen("studentdata.dat","r+");

fp2=fopen("copy2.dat","w+");

if(fp1==NULL)

{

printf("Error in opening a file.");

}

char ids[20];

printf("Enter the student Roll No you want to update:");

cin>>ids;

int i=0;

while(fread(&s[i],sizeof(struct student),1,fp1)){

if(strcmp(s[i].rollno,ids))

{

fwrite(&s[i],sizeof(struct student),1,fp2);

}

i++;

}

struct student s2;

cout<<"\t\t\t\tADD STUDENT"<<endl;

cout<<"\t\t\t\t\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<" Enter Student name:";

cin>>s2.name;

cout<<" Enter Student Phone no:";

cin>>s2.phoneno;

cout<<" Enter Roll no:";

cin>>s2.rollno;

cout<<" Enter Student course:";

cin>>s2.course;

cout<<" Enter Course branch:";

cin>>s2.branch;

cout<<" Enter student year:";

cin>>s2.year;

fwrite(&s2,sizeof(struct student),1,fp2);

fclose(fp1);

fclose(fp2);

remove("studentdata.dat");

rename("copy2.dat","bookstudent.dat");

printf("Record updated successfully.");

displaybook();

}

void updatebook()

{

FILE \*fp1,\*fp2;

fp1=fopen("bookdata.dat","r+");

fp2=fopen("copy.dat","w+");

if(fp1==NULL)

{

printf("Error in opening a file.");

}

char ids[20];

printf("Enter the book ID you want to update:");

cin>>ids;

int i=0;

while(fread(&books[i],sizeof(struct book),1,fp1)){

if(strcmp(books[i].bookid,ids))

{

fwrite(&books[i],sizeof(struct book),1,fp2);

}

i++;

}

struct book s2;

cout<<"\t\t\t\tADD BOOK"<<endl;

cout<<"\t\t\t\t\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<" Enter book id:";

cin>>s2.bookid;

cout<<" Enter book name:";

cin>>s2.bookname;

cout<<" Enter book author:";

cin>>s2.bookauthor;

cout<<" Enter book issue date:";

cin>>s2.bookpubdate;

cout<<" Enter book issue month:";

cin>>s2.bookpubmonth;

cout<<" Enter book year:";

cin>>s2.bookpubyear;

fwrite(&s2,sizeof(struct book),1,fp2);

fclose(fp1);

fclose(fp2);

remove("bookdata.dat");

rename("copy.dat","bookdata.dat");

printf("Record updated successfully.");

displaybook();

}

void deletebook()

{

FILE \*fp1,\*fp2;

fp1=fopen("bookdata.dat","r+");

fp2=fopen("copy.dat","w+");

if(fp1==NULL)

{

printf("Error in opening a file.");

}

char ids[20];

printf("Enter the book ID you want to delete:");

cin>>ids;

int i=0;

while(fread(&books[i],sizeof(struct book),1,fp1)){

if(strcmp(books[i].bookid,ids))

{

fwrite(&books[i],sizeof(struct book),1,fp2);

}

i++;

}

fclose(fp1);

fclose(fp2);

remove("bookdata.dat");

rename("copy.dat","bookdata.dat");

printf("Record updated successfully.");

displaybook();

}

void searchbook()

{

system("CLS");

FILE \*fp;

fp=fopen("bookdata.dat","r");

if(fp==NULL)

{

printf("File can'nt be open");

exit(1);

}

char keyid[20];

int i=0;

cout<<"\t\t\t\tSEARCH BOOK DETAILS"<<endl;

cout<<"\t\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<cout<<" Enter book id to be searched:";

cin>>keyid;

while(fread(&books[i],sizeof(struct book),1,fp))

{

if(!strcmp(books[i].bookid,keyid))

{

cout<<" book name: "<<books[i].bookname<<endl;

cout<<" book author: "<<books[i].bookauthor<<endl;

cout<<" book issue date: "<<books[i].bookpubdate<<endl;

cout<<" book issue month: "<<books[i].bookpubyear<<endl;

cout<<" book issue year: "<<books[i].bookpubmonth<<endl;

}

i++;

}

cout<<"Enter any key to return.";

getch();

}

void addbook()

{

FILE \*fp;

fp=fopen("bookdata.dat","w");

if(fp==NULL)

{

printf("File can'nt be open");

exit(1);

}

int i=0;

int n=0;

char ch;

char id[5],year[4];

while(1)

{

system("CLS");

cout<<"\t\t\t\t\tADD BOOK"<<endl;

cout<<"\t\t\t\t\t\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<" Enter book id:";

cin>>books[i].bookid;

cout<<" Enter book name:";

cin>>books[i].bookname;

cout<<" Enter book author:";

cin>>books[i].bookauthor;

cout<<" Enter book issue date:";

cin>>books[i].bookpubdate;

cout<<" Enter book issue month:";

cin>>books[i].bookpubmonth;

cout<<" Enter book year:";

cin>>books[i].bookpubyear;

fwrite(&books[i],sizeof(struct book),1,fp);

i++;

n++;

cout<<endl<<endl;

printf("Record added successfully.");

cout<<endl<<"Want to add more books??[Y/N]: ";

cin>>ch;

if(ch=='N' || ch=='n')

{

system("CLS");

break;

}

else

{

cout<<"Enter given characters only:";

}

}

fclose(fp);

return;

}

int displaybook()

{

system("CLS");

int i=0;

FILE \*fp;

fp=fopen("bookdata.dat","r");

if(fp==NULL)

{

printf("File can'nt be open");

exit(1);

}

cout<<"\t\t\t\tYour books are"<<endl;

cout<<"\t\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

while(fread(&books[i],sizeof(struct book),1,fp))

{

cout<<" book id: "<<books[i].bookid<<endl;

cout<<" book name: "<<books[i].bookname<<endl;

cout<<" book author: "<<books[i].bookauthor<<endl;

cout<<" book issue date: "<<books[i].bookpubdate<<endl;

cout<<" book issue month: "<<books[i].bookpubyear<<endl;

cout<<" book issue year: "<<books[i].bookpubmonth<<endl<<endl;

i++;

}

fclose(fp);

cout<<"Enter any key to return.";

getch();

}

void library()

{

while(1)

{

system("CLS");

cout<<"\t\t\t\tLIBRARY MANAGEMENT";

cout<<endl;

cout<<"\t\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\t\tchoose the option below"<<endl;

cout<<"\t\t\t\t1: ADD A BOOK"<<endl;

cout<<"\t\t\t\t2: SEARCH ANY BOOKS"<<endl;

cout<<"\t\t\t\t3: DISPLAY ALL BOOKS"<<endl;

cout<<"\t\t\t\t4: DELETE A BOOK"<<endl;

cout<<"\t\t\t\t5: UPDATE BOOK INFORMATION"<<endl;

cout<<"\t\t\t\t6: QUIT"<<endl<<endl<<endl;

cout<<"\t\t\t\tEnter your choice here: ";

char no;

cin>>no;

switch(no)

{

case '1':

addbook();

break;

case '2':

searchbook();

break;

case '3':

displaybook();

break;

case '4':

deletebook();

break;

case '5':

updatebook();

break;

case '6':

exit(1);

default:

{

system("CLS");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"You have entered a wrong key.";

cout<<"Enter any button to reinput your choice:";

char c;

cin>>c;

}

system("CLS");

cout<<endl<<" Returning back to menu.";

sleep(2);

continue;

}

}

}

void studentmanage()

{

while(1)

{

system("CLS");

cout<<"\t\t\t\tSTUDENT MANAGEMENT SYSTEM";

cout<<endl;

cout<<"\t\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\t\tchoose the option below"<<endl;

cout<<"\t\t\t\t1: ADD STUDENT RECORD"<<endl;

cout<<"\t\t\t\t2: SEARCH STUDENT RECORD"<<endl;

cout<<"\t\t\t\t3: DISPLAY ALL RECORD"<<endl;

cout<<"\t\t\t\t4: DELETE A RECORD"<<endl;

cout<<"\t\t\t\t5: UPDATE ANY RECORD"<<endl;

cout<<"\t\t\t\t6: QUIT"<<endl<<endl<<endl;

cout<<"\t\t\t\tEnter your choice here: ";

char no;

cin>>no;

switch(no)

{

case '1':

addrecord();

break;

case '2':

searchrecord();

break;

case '3':

displayrecord();

break;

case '4':

deleterecord();

break;

case '5':

updaterecord();

break;

case '6':

exit(1);

default:

{

system("CLS");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"You have entered a wrong key.";

cout<<"Enter any button to reinput your choice:";

char c;

cin>>c;

}

system("CLS");

cout<<endl<<"\t\t\t\tReturning back to menu.";

sleep(2);

continue;

}

}

}

int main()

{

int no;

char str[20],pass[20];

string username="aditya",password="password";

cout<<endl<<"\t\t\tWELCOME TO MAHARANA PRATAP ENGINEERING COLLEGE"<<endl;

cout<<"\t\t\t\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\t\tLogin:";

cin>>str;

cout<<endl<<"\t\t\t\tPassword:";

cin>>pass;

if(str==username && password==pass)

{

system("CLS");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<"\t\t\t\tYou have succefully login";

sleep(2);

while(1)

{

system("CLS");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

printf("\t\t\t\tChoose any option");

cout<<endl;

cout<<"\t\t\t\t \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<"\t\t\t\t1: Library management"<<endl;

cout<<"\t\t\t\t2: Student record management system"<<endl;

cout<<endl<<endl<<endl;

cout<<"\t\t\t\tEnter your option here:";

cin>>no;

switch(no)

{

case 1:

{

system("CLS");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\tWait for a while we are processing your data!!";

sleep(1);

system("CLS");

library();

break;

}

case 2:

{

system("CLS");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\tWait for a while we are processing your data!!";

sleep(1);

cout<<"2";

system("CLS");

studentmanage();

break;

}

default:

{

system("CLS");

cout<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\t\tSorry you have entered a wrong choice.";

cout<<endl<<endl<<"\t\t\t\tEnter any button to reinput your choice: ";

getch();

}

system("CLS");

cout<<endl<<"\t\t\t\tReturning back to menu.";

sleep(1);

continue;

}

break;

}

}

else

{

cout<<"SORRY YOUR ENTERED WRONG LOGIN Id OR PASSWORD";

}

}

//BOOK Class IN NEW FILE AS **book.cpp**

class book

{

public:

char bookid[10];

char bookname[20];

char bookauthor[20];

char bookpubdate[10];

char bookpubmonth[10];

char bookpubyear[10];

};

///STUDENT Class IN NEW FILE AS student.cpp

class student

{

public:

char name[20];

char phoneno[15];

char rollno[15];

char course[10];

char branch[20];

char year[10];

};