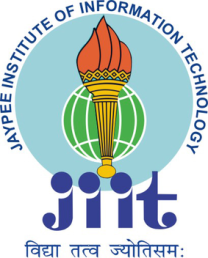
***JAYPEE INSTITUTE OF INFORMATION***

***TECHNOLOGY, NOIDA , SECTOR-62***

******

***SDF-1 MINI PROJECT***

***PROJECT DOCUMENTATION***

***Library management system***

***SUBMITTED TO:***

***Mrs. Dr HEMA N maam***

***BY :***

***TEAM MEMBERS (B10 BATCH):***

* ***HIMANSHU DIXIT (B64178)***
* ***ARYAN RAZDAN (BJ61567)***
* ***AASHUTOSH PRADHAN (B61308)***
* ***ARYAMAN VISHNOI (B63816)***

***(BTECH – CSE 1st SEMESTER)***

***content***

* ***Acknowledgement***
* ***Abstract***
* ***Introduction***
* ***Background of the project***
* ***Objective of the project***
* ***Topics of sdf used***
* ***Modules used***
* ***Project category***
* ***Design of the project and outputs***
* ***implementation***
* ***Conclusion and future scope***
* ***Refernces***

***acknowledgement***

*We extend our sincere thanks to* ***Jaypee Institute of Information and Technology*** *which provided us with the opportunity to make the project using**C Programming which helped us implement and practice skills we have**learned till now.We also take this opportunity to express a deep sense of gratitude to* ***Mrs.Dr. HEMA N Maam*** *for her cordial support, valuable suggestions and**guidance.We would like to thank our friends and family for the support and**encouragement they have given us during the course of work.*

* *HIMANSHU DIXIT (B64178)*
* *ARYAN RAZDAN (BJ61567)*
* *AASHUTOSH PRADHAN (B61308)*
* *ARYAMAN VISHNOI (B63816)*

***ABSTRACT OF THE PROJECT***

*In this project, we are focusing on the concepts of C programming to create a hotel management system .This project uses C language to program the logic of the said management system and create a user-friendly interface. We are using the topics that we have studied in sdf .*

*It is also used to ease the task of librarian.*

***introduction***

*Library Management System is a software used to manages the catalog of a library. This helps to keep the records of whole transactions of the books available in the library. Library Management System which is very easy to use and fulfills all the requirement of a librarian. There are many features which helps librarian to keep records of available books as well as issued books.*

*Here is a list of some features of Library Management System.*

* *Keep record of different categories like; Books, Journals, Newspapers,Magazines, etc.*
* *Classify the books subject wise.*
* *Easy way to enter new books.*
* *Keep record of complete information of a book like; Book name, Author name,Publisher’s name, Date/ Year of publication, Cost of the book, Book purchasingdate/ Bill no.*
* *Easy way to make a check-out.*
* *Easy way to make a check-in.*
* *Automatic fine calculation for late returns.*
* *Different criteria for searching a book.*
* *Different kind of reports like; total no. of books, no. of issued books, no. of journals, etc.*
* *Easy way to know how many books are issued to a particular student.*
* *Easy way to know the status of a book and much more.*

***Background of the project***

*E-Library Management System is an application which refers to library systems which are generally small or medium in size. It is used by librarian to manage the library using a computerized system where he/she can add new books, videos and Page sources. Books and student maintenance modules are also included in this system which would keep track of the students using the library and also a detailed description about the books a library contains. With this computerized system there will be no loss of book record or member record which generally happens when a non -computerized system is used. All these modules are able to help librarian to manage the library with more convenience and in a more efficient way as compared to library systems which are not computerized.*

***objective of the project***

*Library Management System is a system which maintains the information about the books present in the library, their authors, the members of library to whom books are issued, library staff and all. This is very difficult to organize manually. Maintenance of all this information manually is a very complex task. Owing to the advancement of technology, organization of an Online Library becomes much simple. The Online Library Management has been designed to computerize and automate the operations performed over the information about the members, book issues and returns and all other operations. This computerization of library helps in many instances of its maintenances. It reduces the workload of management as most of the manual work done is reduced. The system deals with the all the requirements needed for managing the library. It deals with the process of managing data about the books in the library. Library management System maintains the record of books in the, issue and return process of the book in the library. Here we are primarily concerned with management of books of library. In this project we identify the need for computer based Library Management System.*

***topics of sdf used***

* *Basis of c language.*
* *Data types*
* *Operators*
* *If else*
* *Arrays*
* *Function*
* *Structure*
* *Loops*
* *Data file handling*
* *Pointers*

***modules used***

*The various modules used in the projects are:*

* *Issue Book*
* *Return Book*
* *Search Book*
* *Deleting Book.*
* *Add Book.*
* *Update password.*

***Note- New Modules and Sub Modules can be added as per the requirement of the management system.***

***project category***

*This Project comes under file handling management category in C language. File*

*handling is done using concept of array in c language. It is console based and user interactive.The UI is interactive and easily understandable.*

*Software used:*

* *Software apps like Code blocks or visual studio code are used in creating in the project.*
* *Code blocks: Is an IDE used for C language. It has inbuilt compiler.*
* *Visual Studio Code: It is same like Code blocks but a compiler like mingw or any has to be installed for compilation of the code.*
* *System can be any 32/62 bit.*
* *Ram should be greater than or equal to 2 GB.*

***Design of the project /***

***outputs***

Print the intro

Print the wel -come screen

Declare and read two string userid and pwd

Its check the userid and password if not correct it will again give a chance to enter userid and pwd.

Print the main menu with 4 option given below

Declare choice and take input

Book issue function() calls if choice=1

Exit if choice=0

Updatepwd function() calls if choice=4

Administrator menu function() calls if choice=3

Book deposit function() calls if choice=2

*if enter invalid choice*

Declare choice and take input

Exit if choice=0

Update pwd function() calls if choice=7

Issue Book function() calls if choice=5

View Book function() calls if choice=3

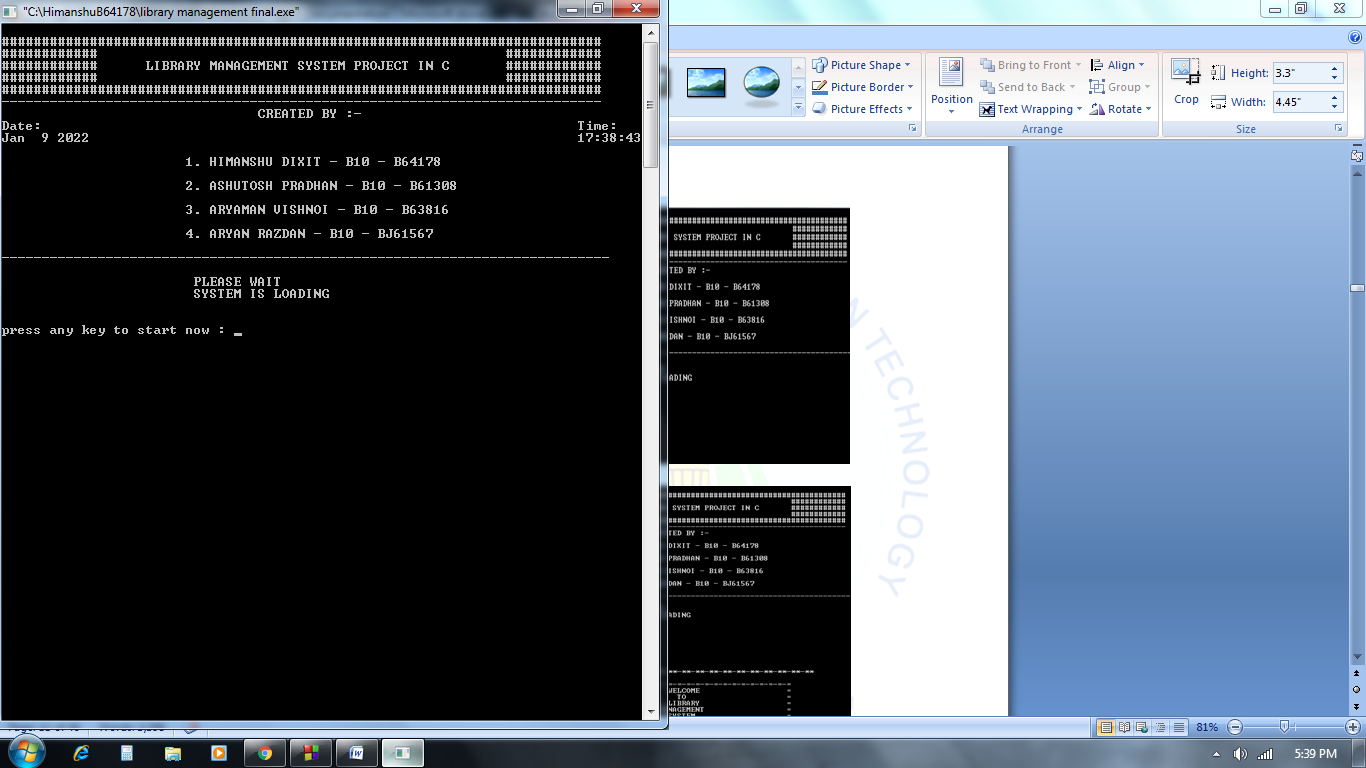
Add Book function() calls if choice=1

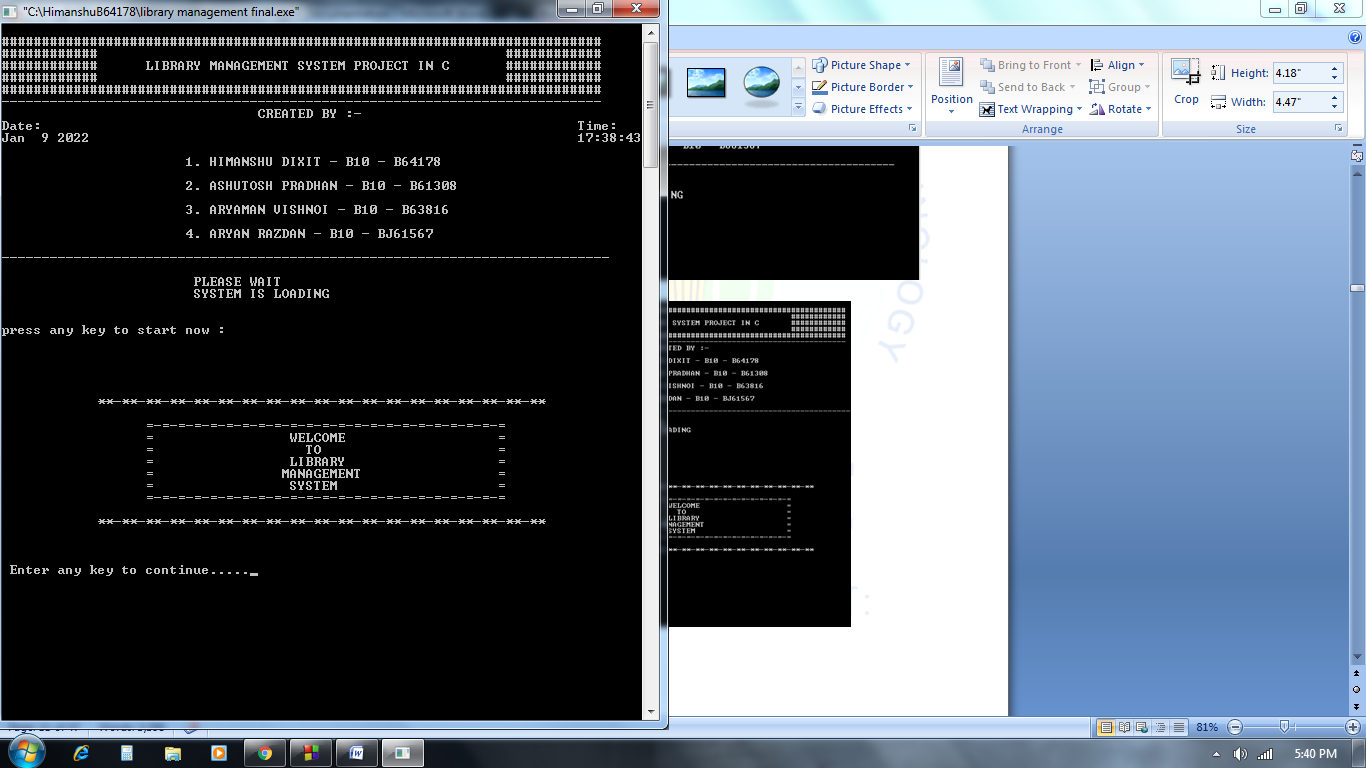
Back if choice=8

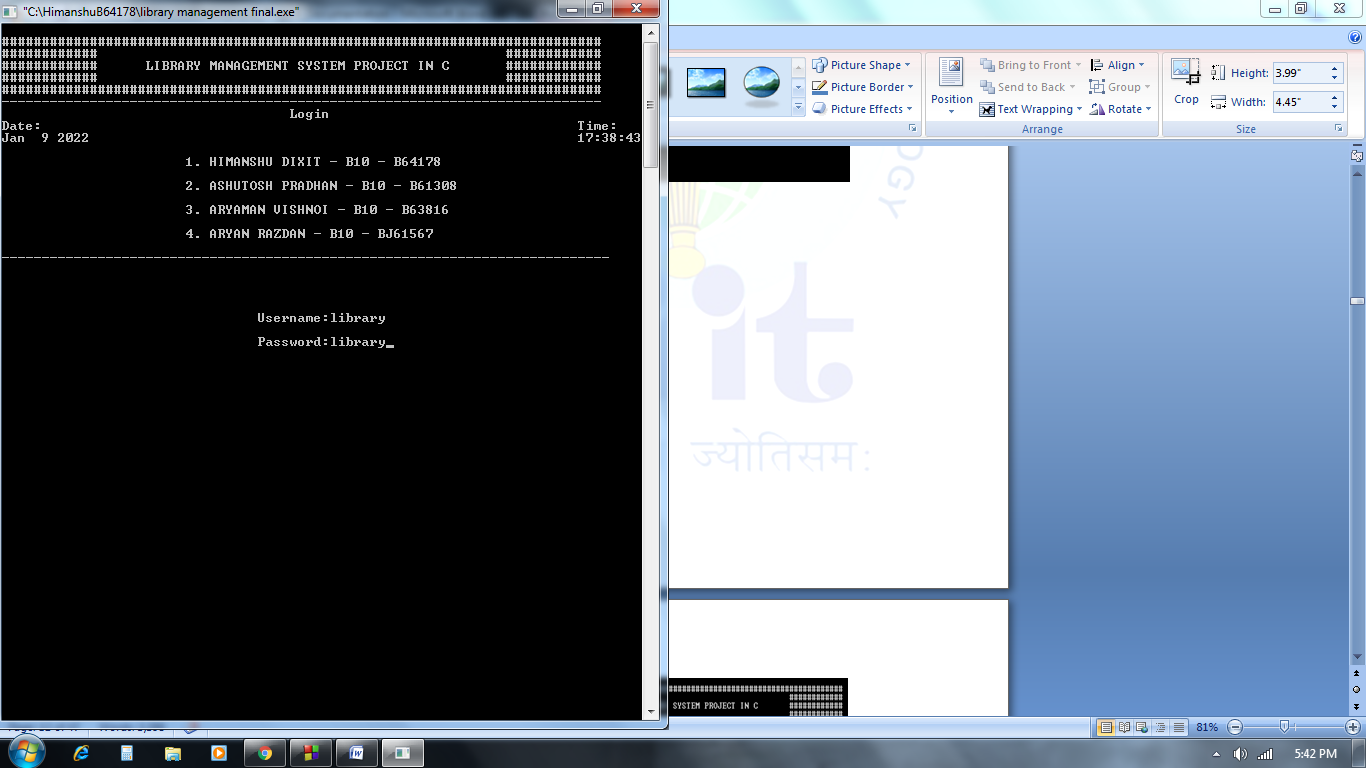
Deposit Book function() calls if choice=6

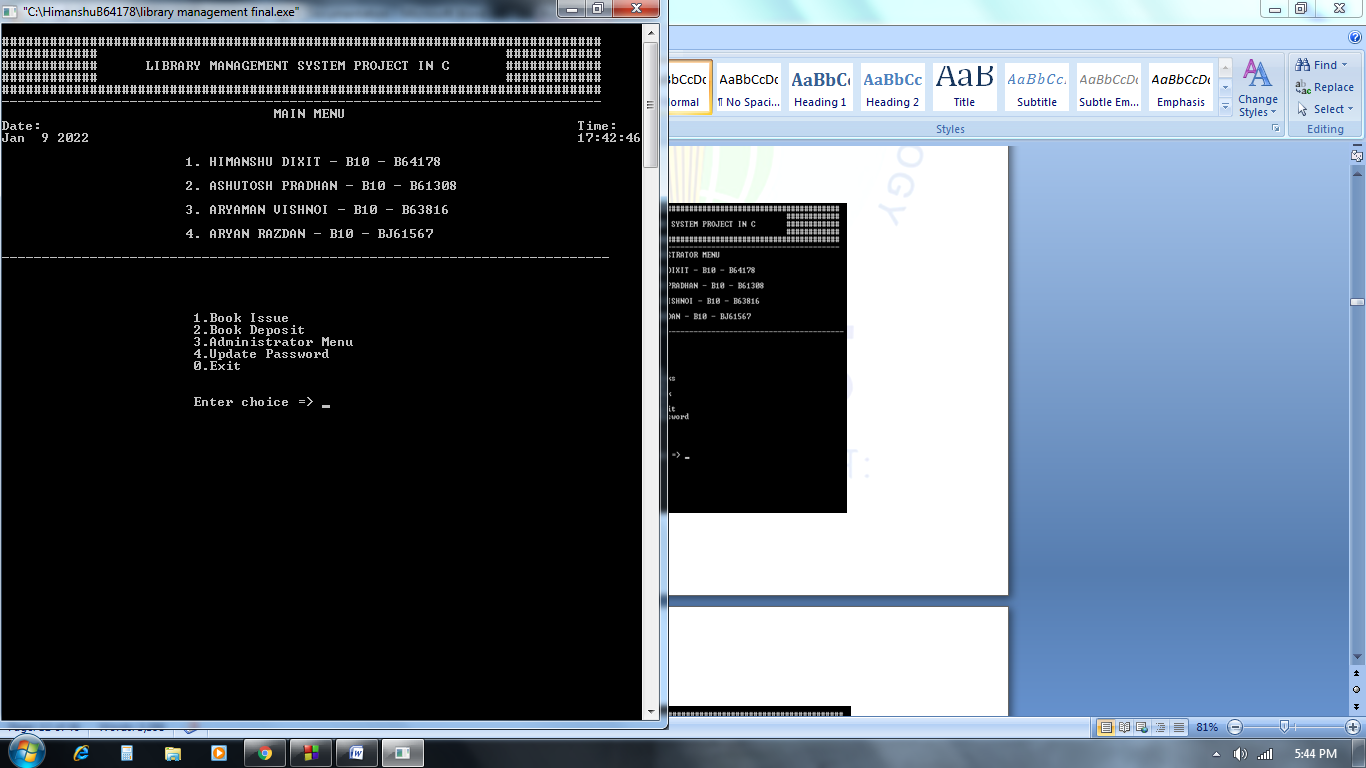
Delete Book function() calls if choice=4

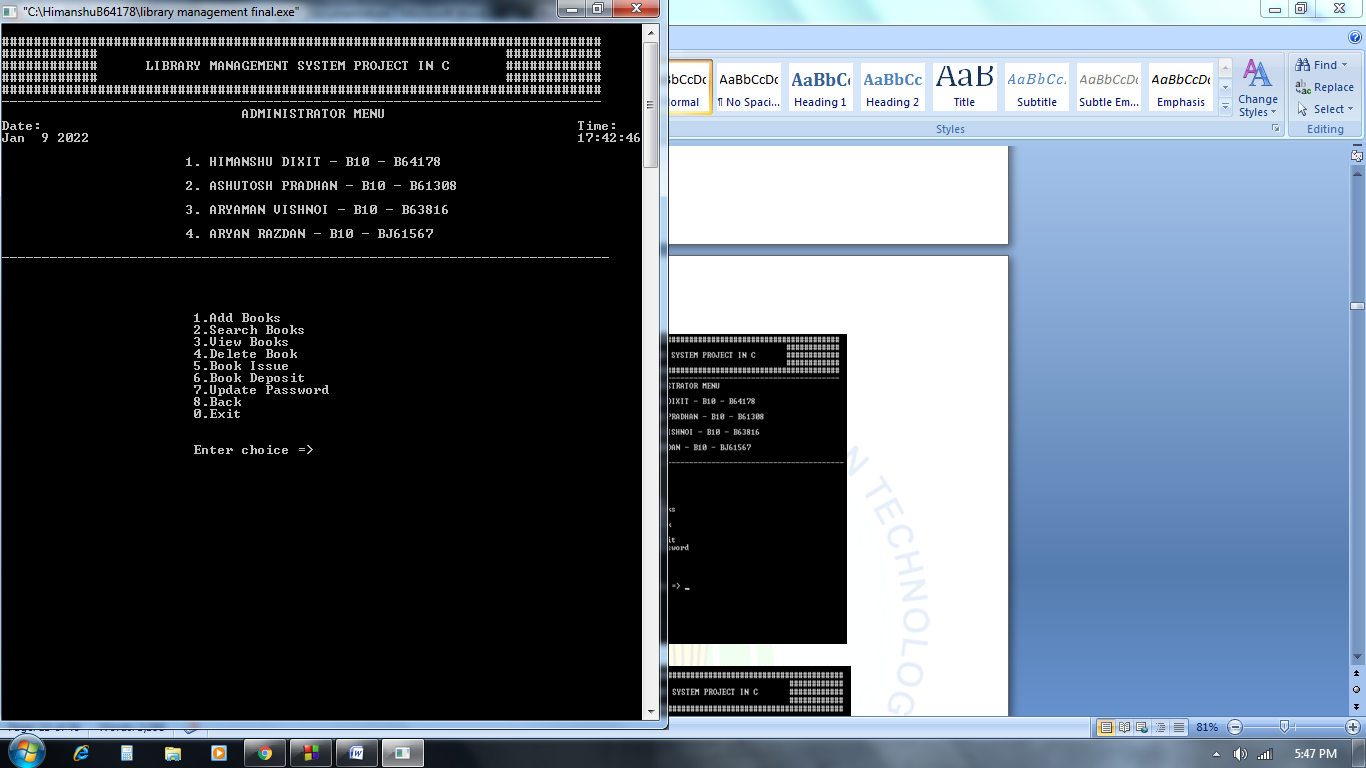
Search Book function() calls if choice=2

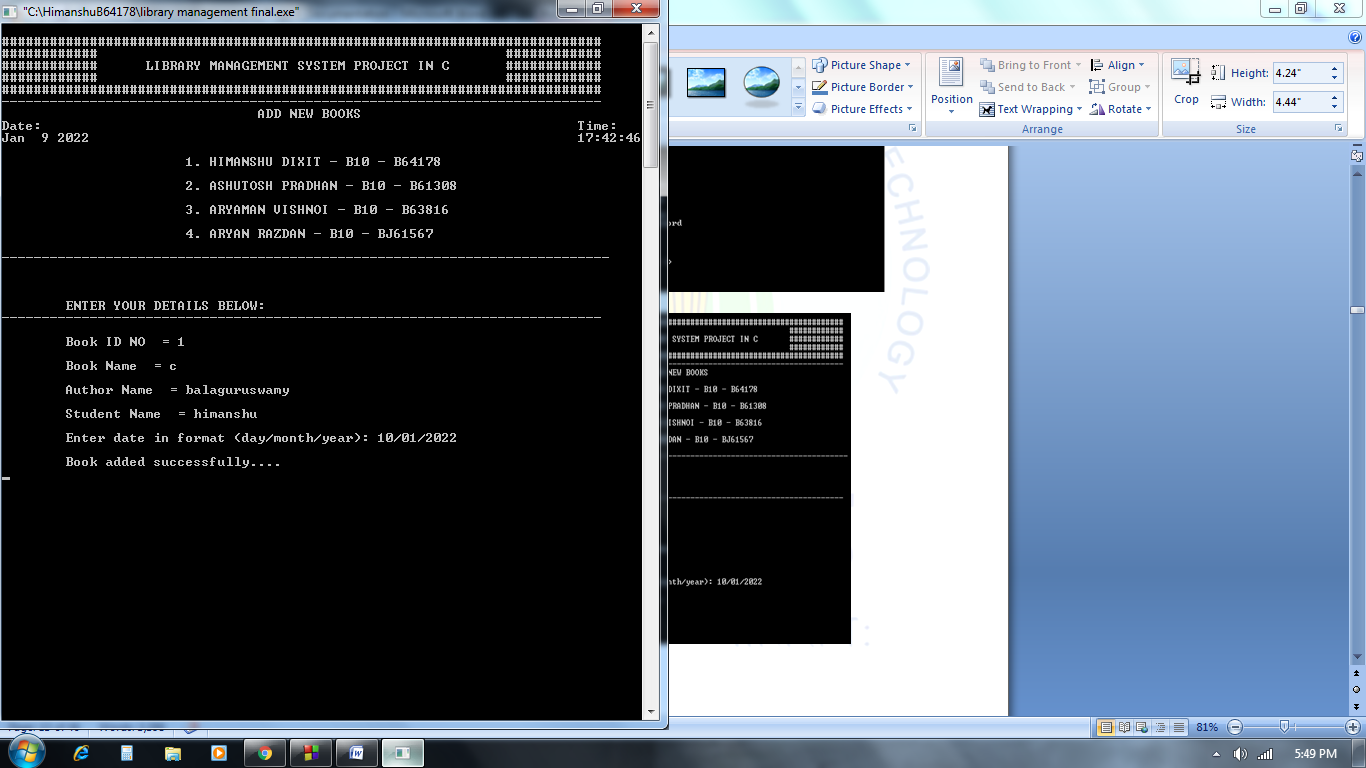
**

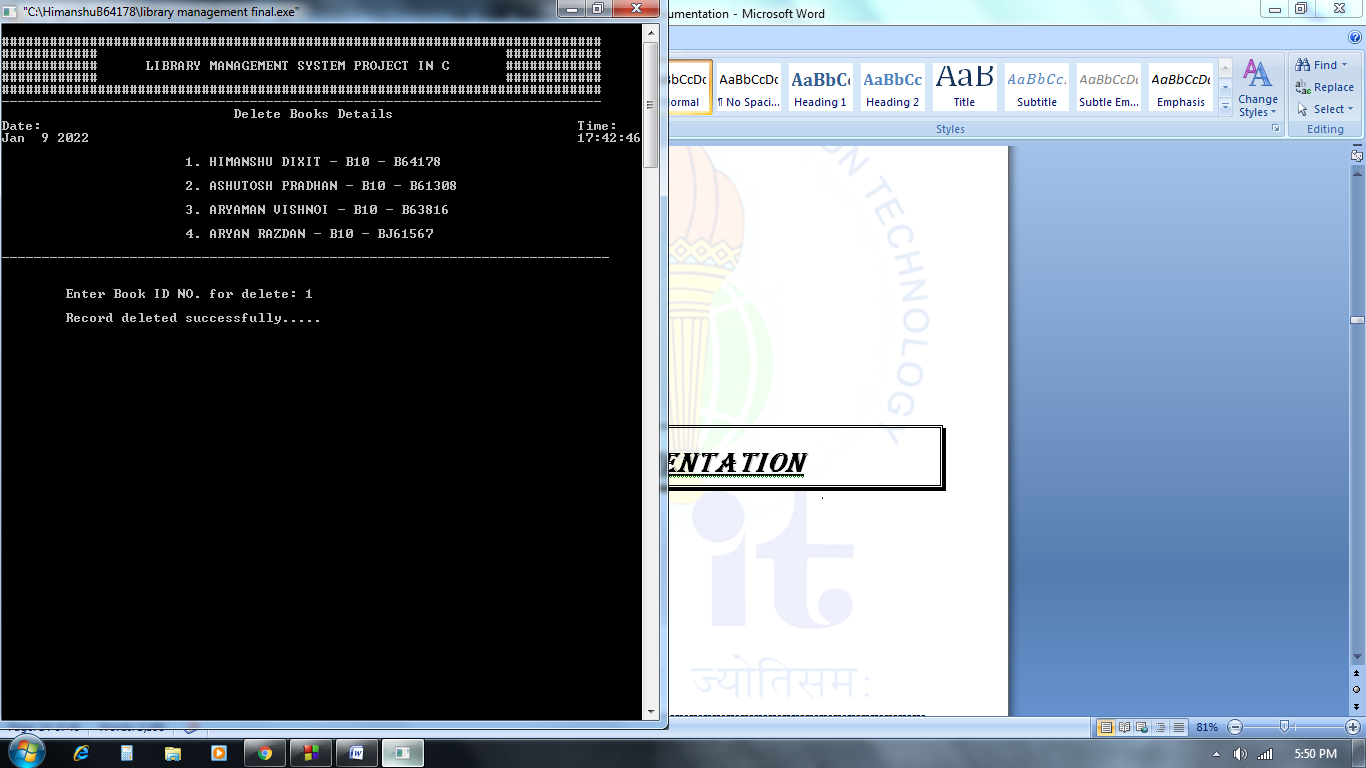
**

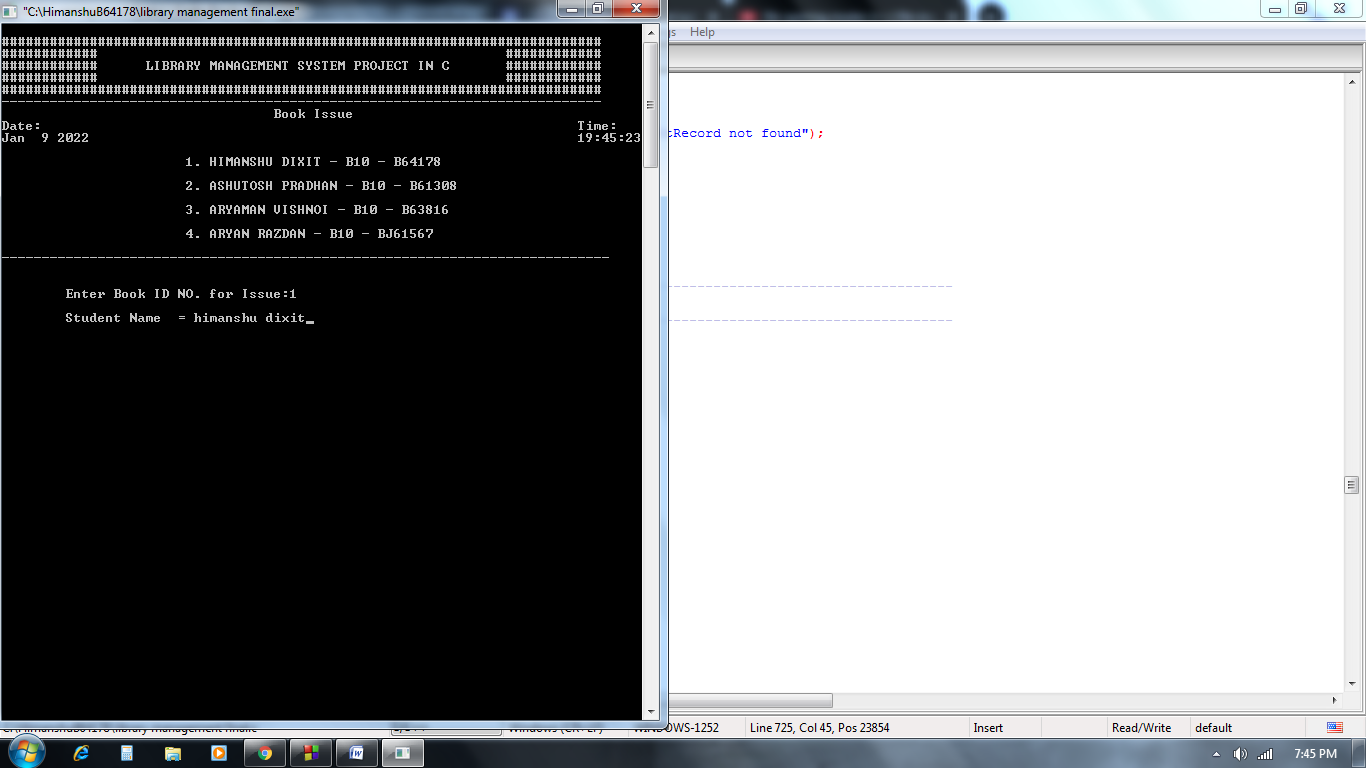
**

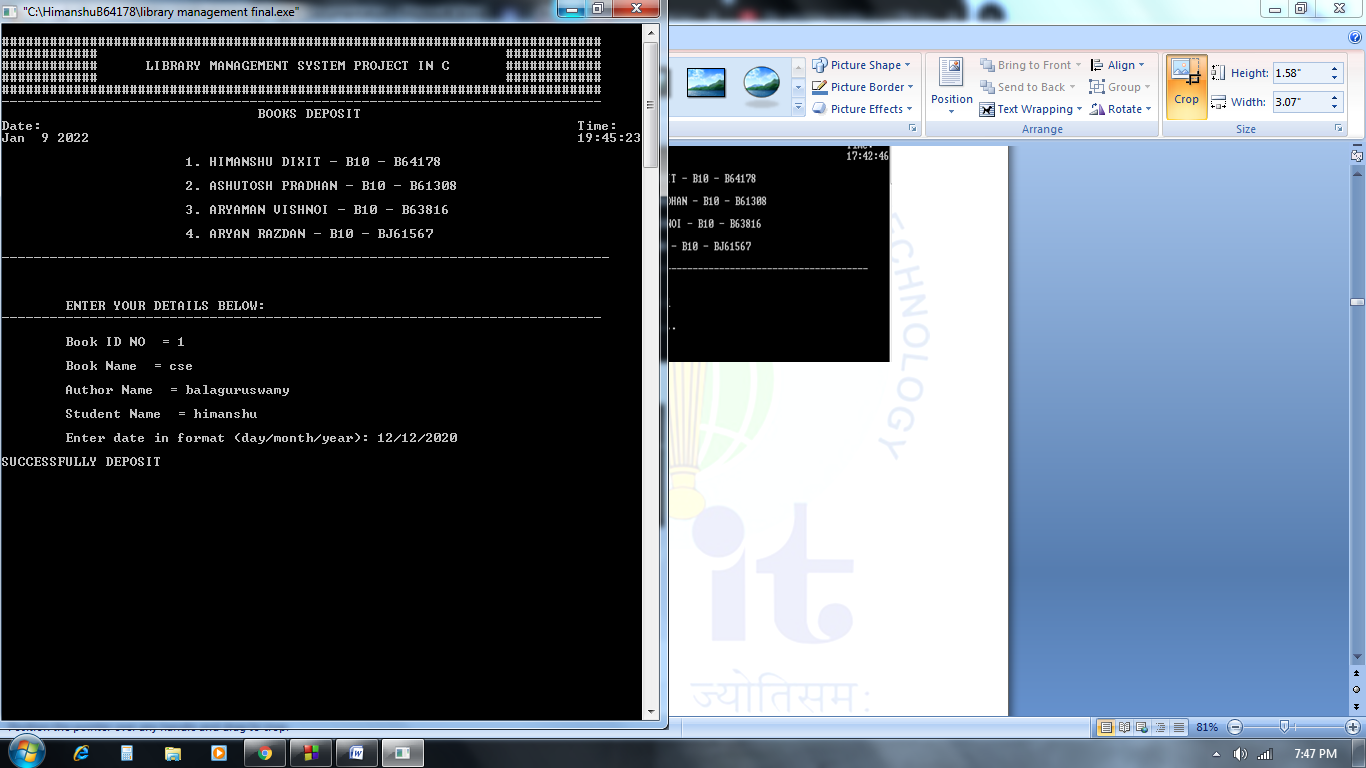
**

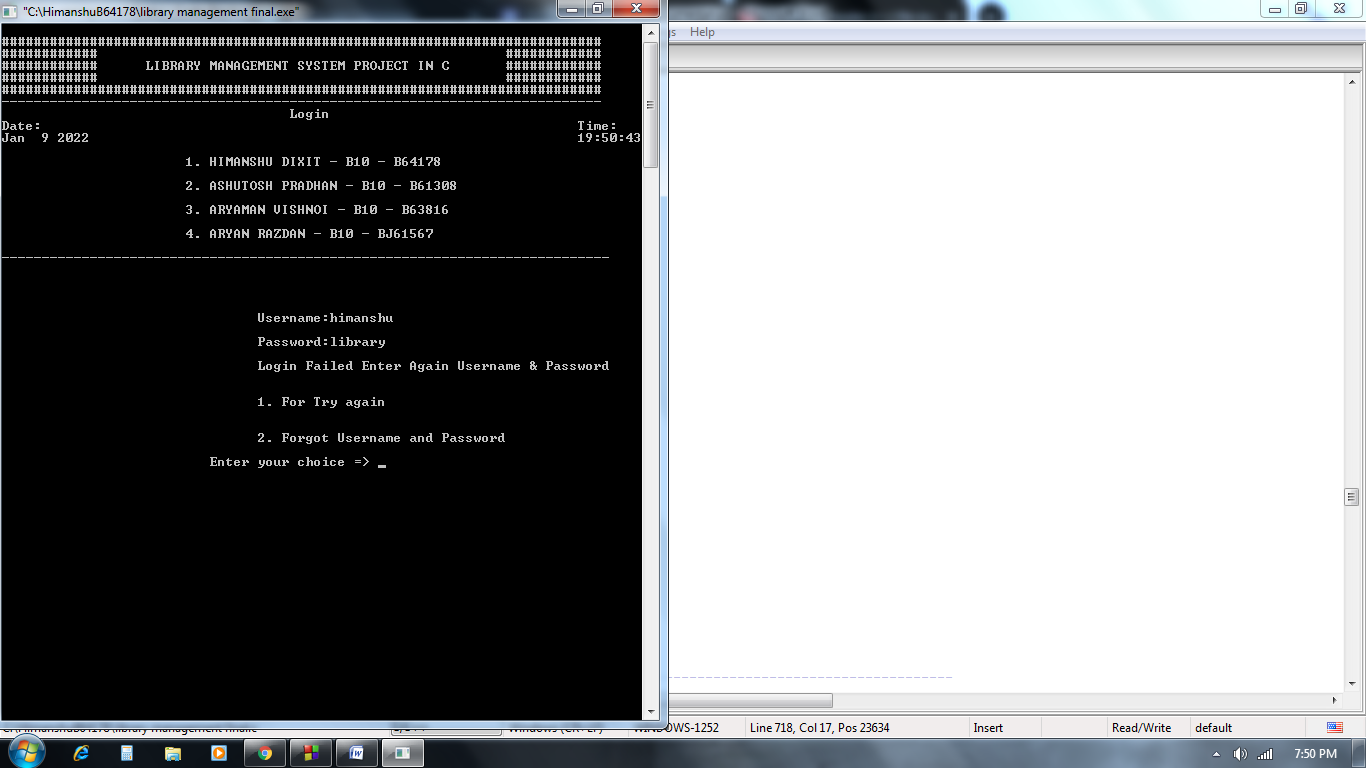
**

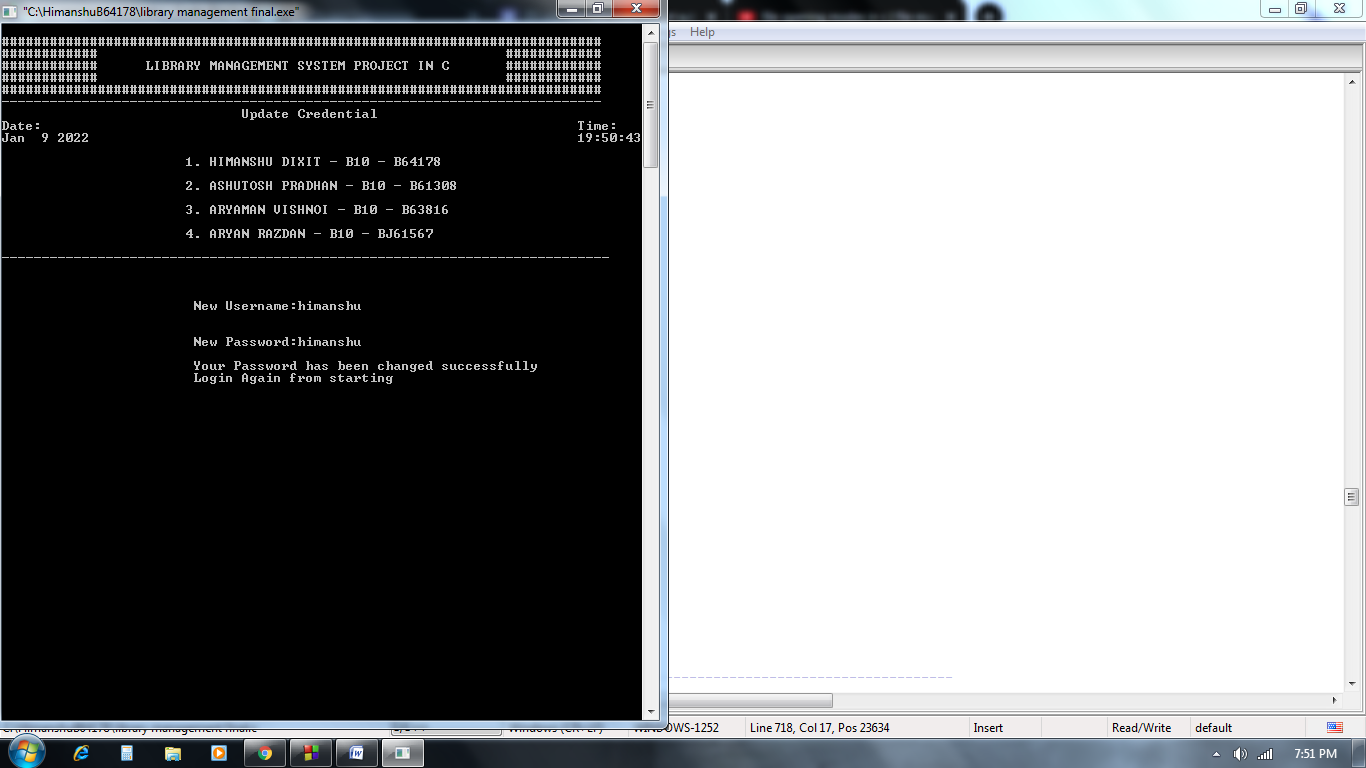
**

**

**

**

**

**

***implementation***

*#include <stdio.h>*

*#include <time.h>*

*#include <string.h>*

*#include <stdlib.h>*

*#include <conio.h>*

*//---------------------------------------------------------------------------------------------------------*

*#define MAX\_YR 9999*

*#define MIN\_YR 1900*

*#define MAX\_SIZE\_USER\_NAME 30*

*#define MAX\_SIZE\_PASSWORD 20*

*//---------------------------------------------------------------------------------------------------------*

*#define FILE\_NAME "library.bin"*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*// Macro related to the books info*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*#define MAX\_BOOK\_NAME 50*

*#define MAX\_AUTHOR\_NAME 50*

*#define MAX\_STUDENT\_NAME 50*

*#define MAX\_STUDENT\_ADDRESS 300*

*#define FILE\_HEADER\_SIZE sizeof(sFileHeader)*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//structure to store date*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*typedef struct*

*{*

*int yyyy;*

*int mm;*

*int dd;*

*} Date;*

*//---------------------------------------------------------------------------------------------------------*

*typedef struct*

*{*

*char username[MAX\_SIZE\_USER\_NAME];*

*char password[MAX\_SIZE\_PASSWORD];*

*} sFileHeader;*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*// to call in program*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*typedef struct*

*{*

*// declare the integer data type*

*unsigned int books\_id;*

*// declare the character data type*

*char bookName[MAX\_BOOK\_NAME];*

*// declare the character data type*

*char authorName[MAX\_AUTHOR\_NAME];*

*// declare the charecter data type*

*char studentName[MAX\_STUDENT\_NAME];*

*// declare the character data type*

*char studentAddr[MAX\_STUDENT\_ADDRESS];*

*// declare the integer data type*

*Date bookIssueDate;*

*} s\_BooksInfo;*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//print the group members name*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*void printMessageCenter(const char\* message)*

*{*

*int len =0;*

*int pos = 0;*

*//calculate how many space need to print*

*len = (78 - strlen(message))/2;*

*//printf("\t\t\t");*

*for(pos =0 ; pos < len ; pos++)*

*{*

*//print space*

*printf(" ");*

*}*

*//print message*

*printf("%s",message);*

*printf("\nDate:\t\t\t\t\t\t\t\t\tTime:\t%s\t\t\t\t\t\t\t\t%s",\_\_DATE\_\_,\_\_TIME\_\_);*

*printf("\n\t\t 1. HIMANSHU DIXIT - B10 - B64178");*

*printf("\n\n\t\t 2. ASHUTOSH PRADHAN - B10 - B61308");*

*printf("\n\n\t\t 3. ARYAMAN VISHNOI - B10 - B63816");*

*printf("\n\n\t\t 4. ARYAN RAZDAN - B10 - BJ61567\n");*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//print the head message always show in above*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*void headMessage(const char \*message)*

*{*

*system("cls");*

*printf("\n###########################################################################");*

*printf("\n############ ############");*

*printf("\n############ LIBRARY MANAGEMENT SYSTEM PROJECT IN C ############");*

*printf("\n############ ############");*

*printf("\n###########################################################################");*

*printf("\n---------------------------------------------------------------------------\n");*

*printMessageCenter(message);*

*printf("\n----------------------------------------------------------------------------\n\n");*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//loading page after loading press key to start*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*void loadingpage()*

*{*

*printf("\t\t\tPLEASE WAIT\n");*

*printf("\t\t\tSYSTEM IS LOADING\n");*

*printf("\n\npress any key to start now : ");*

*getch();*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//function to print the welcome message*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*void welcomeMessage()*

*{*

*headMessage("CREATED BY :-");*

*loadingpage();*

*printf("\n\n\n\n\n");*

*printf("\n\t \*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*\n");*

*printf("\n\t =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=");*

*printf("\n\t = WELCOME =");*

*printf("\n\t = TO =");*

*printf("\n\t = LIBRARY =");*

*printf("\n\t = MANAGEMENT =");*

*printf("\n\t = SYSTEM =");*

*printf("\n\t =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=");*

*printf("\n\n\t \*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*-\*\*\n");*

*printf("\n\n\n Enter any key to continue.....");*

*getch();*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//function to check name is valid has alphabet or not*

*//---------------------------------------------------------------------------------------------------------*

*int check(char a)*

*{*

*if(a>='a' && a<='z' || a>='A' && a<='Z')*

*return 1;*

*else*

*return 0;*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//function to check name*

*//---------------------------------------------------------------------------------------------------------*

*int isNameValid(const char \*name)*

*{*

*int validName = 1;*

*int len = 0;*

*int index = 0;*

*len = strlen(name);*

*for(index =0; index <len ; ++index)*

*{*

*int l = check(name[index]);*

*if((l == 0) && (name[index] != '\n') && (name[index] != ' '))*

*{*

*validName = 0;*

*break;*

*}*

*}*

*return validName;*

*}*

*//---------------------------------------------------------------------------------------------------------*

*// Function to check leap year.*

*//Function returns 1 if leap year*

*//---------------------------------------------------------------------------------------------------------*

*int IsLeapYear(int year)*

*{*

*return (((year % 4 == 0) &&*

*(year % 100 != 0)) ||*

*(year % 400 == 0));*

*}*

*//---------------------------------------------------------------------------------------------------------*

*// returns 1 if given date is valid.*

*//---------------------------------------------------------------------------------------------------------*

*int isValidDate(Date \*validDate)*

*{*

*//check range of year,month and day*

*if (validDate->yyyy > MAX\_YR ||*

*validDate->yyyy < MIN\_YR)*

*return 0;*

*if (validDate->mm < 1 || validDate->mm > 12)*

*return 0;*

*if (validDate->dd < 1 || validDate->dd > 31)*

*return 0;*

*//Handle feb days in leap year*

*if (validDate->mm == 2)*

*{*

*if (IsLeapYear(validDate->yyyy))*

*return (validDate->dd <= 29);*

*else*

*return (validDate->dd <= 28);*

*}*

*//handle months which has only 30 days*

*if (validDate->mm == 4 || validDate->mm == 6 ||*

*validDate->mm == 9 || validDate->mm == 11)*

*return (validDate->dd <= 30);*

*return 1;*

*}*

*//---------------------------------------------------------------------------------------------------------*

*// Add books in list*

*//---------------------------------------------------------------------------------------------------------*

*void addBookInDataBase()*

*{*

*int days;*

*s\_BooksInfo addBookInfoInDataBase = {0};*

*FILE \*fp = NULL;*

*int status = 0;*

*fp = fopen(FILE\_NAME,"a+");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*headMessage("ADD NEW BOOKS");*

*printf("\n\n\tENTER YOUR DETAILS BELOW:");*

*printf("\n---------------------------------------------------------------------------\n");*

*printf("\n\tBook ID NO = ");*

*fflush(stdin);*

*scanf("%u",&addBookInfoInDataBase.books\_id);*

*do*

*{*

*printf("\n\tBook Name = ");*

*fflush(stdin);*

*fgets(addBookInfoInDataBase.bookName,MAX\_BOOK\_NAME,stdin);*

*status = isNameValid(addBookInfoInDataBase.bookName);*

*if (!status)*

*{*

*printf("\n\tName contain invalid character. Please enter again.");*

*}*

*}*

*while(!status);*

*do*

*{*

*printf("\n\tAuthor Name = ");*

*fflush(stdin);*

*fgets(addBookInfoInDataBase.authorName,MAX\_AUTHOR\_NAME,stdin);*

*status = isNameValid(addBookInfoInDataBase.authorName);*

*if (!status)*

*{*

*printf("\n\tName contain invalid character. Please enter again.");*

*}*

*}*

*while(!status);*

*do*

*{*

*//get date year,month and day from user*

*printf("\n\tEnter date in format (day/month/year): ");*

*scanf("%d/%d/%d",&addBookInfoInDataBase.bookIssueDate.dd,&addBookInfoInDataBase.bookIssueDate.mm,&addBookInfoInDataBase.bookIssueDate.yyyy);*

*//check date validity*

*status = isValidDate(&addBookInfoInDataBase.bookIssueDate);*

*if (!status)*

*{*

*printf("\n\tPlease enter a valid date.\n");*

*}*

*}*

*while(!status);*

*printf("\n\tBook added successfully....\n");*

*getch();*

*fwrite(&addBookInfoInDataBase,sizeof(addBookInfoInDataBase), 1, fp);*

*fclose(fp);*

*}*

*//---------------------------------------------------------------------------------------------------------*

*void sBooks()*

*{*

*int found = 0;*

*char bookName[MAX\_BOOK\_NAME] = {0};*

*s\_BooksInfo addBookInfoInDataBase = {0};*

*FILE \*fp = NULL;*

*int status = 0;*

*fp = fopen(FILE\_NAME,"r");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*headMessage("SEARCH BOOKS");*

*//put the control on books detail*

*if (fseek(fp,FILE\_HEADER\_SIZE,SEEK\_SET) != 0)*

*{*

*fclose(fp);*

*printf("\n\tFacing issue while reading file\n");*

*exit(1);*

*}*

*printf("\n\n\tEnter Book Name to search:");*

*fflush(stdin);*

*fgets(bookName,MAX\_BOOK\_NAME,stdin);*

*while (fread (&addBookInfoInDataBase, sizeof(addBookInfoInDataBase), 1, fp))*

*{*

*if(!strcmp(addBookInfoInDataBase.bookName, bookName))*

*{*

*found = 1;*

*break;*

*}*

*}*

*if(found)*

*{*

*printf("\n\tBook id = %u\n",addBookInfoInDataBase.books\_id);*

*printf("\tBook name = %s",addBookInfoInDataBase.bookName);*

*printf("\tBook authorName = %s",addBookInfoInDataBase.authorName);*

*printf("\tBook issue date(day/month/year) = (%d/%d/%d)",addBookInfoInDataBase.bookIssueDate.dd,*

*addBookInfoInDataBase.bookIssueDate.mm, addBookInfoInDataBase.bookIssueDate.yyyy);*

*}*

*else*

*{*

*printf("\n\tNo Record");*

*}*

*fclose(fp);*

*printf("\n\n\n\tPress any key to go to main menu.....");*

*getchar();*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//book issue*

*//---------------------------------------------------------------------------------------------------------*

*void bookissue()*

*{*

*int found = 0;*

*int bookIssue = 0;*

*sFileHeader fileHeaderInfo = {0};*

*char bookName[MAX\_BOOK\_NAME] = {0};*

*s\_BooksInfo addBookInfoInDataBase = {0};*

*FILE \*fp = NULL;*

*FILE \*tmpFp = NULL;*

*int status = 0;*

*headMessage("Book Issue");*

*fp = fopen(FILE\_NAME,"r");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*tmpFp = fopen("tmp.bin","w");*

*if(tmpFp == NULL)*

*{*

*fclose(fp);*

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

*exit(1);*

*}*

*fread (&fileHeaderInfo,FILE\_HEADER\_SIZE, 1, fp);*

*fwrite(&fileHeaderInfo,FILE\_HEADER\_SIZE, 1, tmpFp);*

*printf("\n\tEnter Book ID NO. for Issue:");*

*scanf("%d",&bookIssue);*

*do*

*{*

*printf("\n\tStudent Name = ");*

*fflush(stdin);*

*fgets(addBookInfoInDataBase.studentName,MAX\_STUDENT\_NAME,stdin);*

*status = isNameValid(addBookInfoInDataBase.studentName);*

*if (!status)*

*{*

*printf("\n\tName contain invalid character. Please enter again.");*

*}*

*}*

*while(!status);*

*while (fread (&addBookInfoInDataBase, sizeof(addBookInfoInDataBase), 1, fp))*

*{*

*if(addBookInfoInDataBase.books\_id != bookIssue)*

*{*

*fwrite(&addBookInfoInDataBase,sizeof(addBookInfoInDataBase), 1, tmpFp);*

*}*

*else*

*{*

*found = 1;*

*}*

*}*

*(found)? printf("\n\tBook Issued successfully....."):printf("\n\tBook not found");*

*getch();*

*fclose(fp);*

*fclose(tmpFp);*

*remove(FILE\_NAME);*

*rename("tmp.bin",FILE\_NAME);*

*}*

*//---------------------------------------------------------------------------------------------------------*

*// book deposit*

*//---------------------------------------------------------------------------------------------------------*

*void bookdeposit()*

*{*

*int days;*

*s\_BooksInfo addBookInfoInDataBase = {0};*

*FILE \*fp = NULL;*

*int status = 0;*

*fp = fopen(FILE\_NAME,"a+");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*headMessage("BOOKS DEPOSIT");*

*printf("\n\n\tENTER YOUR DETAILS BELOW:");*

*printf("\n---------------------------------------------------------------------------\n");*

*printf("\n\tBook ID NO = ");*

*fflush(stdin);*

*scanf("%u",&addBookInfoInDataBase.books\_id);*

*do*

*{*

*printf("\n\tBook Name = ");*

*fflush(stdin);*

*fgets(addBookInfoInDataBase.bookName,MAX\_BOOK\_NAME,stdin);*

*status = isNameValid(addBookInfoInDataBase.bookName);*

*if (!status)*

*{*

*printf("\n\tName contain invalid character. Please enter again.");*

*}*

*}*

*while(!status);*

*do*

*{*

*printf("\n\tAuthor Name = ");*

*fflush(stdin);*

*fgets(addBookInfoInDataBase.authorName,MAX\_AUTHOR\_NAME,stdin);*

*status = isNameValid(addBookInfoInDataBase.authorName);*

*if (!status)*

*{*

*printf("\n\tName contain invalid character. Please enter again.");*

*}*

*}*

*while(!status);*

*do*

*{*

*printf("\n\tStudent Name = ");*

*fflush(stdin);*

*fgets(addBookInfoInDataBase.studentName,MAX\_STUDENT\_NAME,stdin);*

*status = isNameValid(addBookInfoInDataBase.studentName);*

*if (!status)*

*{*

*printf("\n\tName contain invalid character. Please enter again.");*

*}*

*}*

*while(!status);*

*do*

*{*

*//get date year,month and day from user*

*printf("\n\tEnter date in format (day/month/year): ");*

*scanf("%d/%d/%d",&addBookInfoInDataBase.bookIssueDate.dd,&addBookInfoInDataBase.bookIssueDate.mm,&addBookInfoInDataBase.bookIssueDate.yyyy);*

*//check date validity*

*status = isValidDate(&addBookInfoInDataBase.bookIssueDate);*

*if (!status)*

*{*

*printf("\n\tPlease enter a valid date.\n");*

*}*

*}*

*while(!status);*

*printf("\nSUCCESSFULLY DEPOSIT ");*

*getch();*

*fwrite(&addBookInfoInDataBase,sizeof(addBookInfoInDataBase), 1, fp);*

*fclose(fp);*

*}*

*//---------------------------------------------------------------------------------------------------------*

*// search books*

*//---------------------------------------------------------------------------------------------------------*

*void searchBooks()*

*{*

*int found = 0;*

*char bookName[MAX\_BOOK\_NAME] = {0};*

*s\_BooksInfo addBookInfoInDataBase = {0};*

*FILE \*fp = NULL;*

*int status = 0;*

*fp = fopen(FILE\_NAME,"r");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*headMessage("SEARCH BOOKS");*

*//put the control on books detail*

*if (fseek(fp,FILE\_HEADER\_SIZE,SEEK\_SET) != 0)*

*{*

*fclose(fp);*

*printf("\n\tFacing issue while reading file\n");*

*exit(1);*

*}*

*printf("\n\n\tEnter Book Name to search:");*

*fflush(stdin);*

*fgets(bookName,MAX\_BOOK\_NAME,stdin);*

*while (fread (&addBookInfoInDataBase, sizeof(addBookInfoInDataBase), 1, fp))*

*{*

*if(!strcmp(addBookInfoInDataBase.bookName, bookName))*

*{*

*found = 1;*

*break;*

*}*

*}*

*if(found)*

*{*

*printf("\n\tBook id = %u\n",addBookInfoInDataBase.books\_id);*

*printf("\tBook name = %s",addBookInfoInDataBase.bookName);*

*printf("\tBook authorName = %s",addBookInfoInDataBase.authorName);*

*printf("\tBook issue date(day/month/year) = (%d/%d/%d)",addBookInfoInDataBase.bookIssueDate.dd,*

*addBookInfoInDataBase.bookIssueDate.mm, addBookInfoInDataBase.bookIssueDate.yyyy);*

*}*

*else*

*{*

*printf("\n\tNo Record");*

*}*

*fclose(fp);*

*printf("\n\n\n\tPress any key to go to main menu.....");*

*getchar();*

*}*

*//---------------------------------------------------------------------------------------------------------*

*// view books function*

*//---------------------------------------------------------------------------------------------------------*

*void viewBooks()*

*{*

*int found = 0;*

*char bookName[MAX\_BOOK\_NAME] = {0};*

*s\_BooksInfo addBookInfoInDataBase = {0};*

*FILE \*fp = NULL;*

*int status = 0;*

*unsigned int countBook = 1;*

*headMessage("VIEW BOOKS DETAILS");*

*fp = fopen(FILE\_NAME,"r");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*if (fseek(fp,FILE\_HEADER\_SIZE,SEEK\_SET) != 0)*

*{*

*fclose(fp);*

*printf("Facing issue while reading file\n");*

*exit(1);*

*}*

*while (fread (&addBookInfoInDataBase, sizeof(addBookInfoInDataBase), 1, fp))*

*{*

*printf("\n\tBook Count = %d\n\n",countBook);*

*printf("\tBook id = %u",addBookInfoInDataBase.books\_id);*

*printf("\n\tBook name = %s",addBookInfoInDataBase.bookName);*

*printf("\t\tBook authorName = %s",addBookInfoInDataBase.authorName);*

*printf("\tBook issue date(day/month/year) = (%d/%d/%d)\n\n",addBookInfoInDataBase.bookIssueDate.dd,*

*addBookInfoInDataBase.bookIssueDate.mm, addBookInfoInDataBase.bookIssueDate.yyyy);*

*found = 1;*

*++countBook;*

*}*

*fclose(fp);*

*if(!found)*

*{*

*printf("\n\tNo Record");*

*}*

*printf("\n\n\tPress any key to go to main menu.....");*

*fflush(stdin);*

*getchar();*

*}*

*//---------------------------------------------------------------------------------------------------------*

*// delete function*

*//---------------------------------------------------------------------------------------------------------*

*void deleteBooks()*

*{*

*int found = 0;*

*int bookDelete = 0;*

*sFileHeader fileHeaderInfo = {0};*

*char bookName[MAX\_BOOK\_NAME] = {0};*

*s\_BooksInfo addBookInfoInDataBase = {0};*

*FILE \*fp = NULL;*

*FILE \*tmpFp = NULL;*

*int status = 0;*

*headMessage("Delete Books Details");*

*fp = fopen(FILE\_NAME,"r");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*tmpFp = fopen("tmp.bin","w");*

*if(tmpFp == NULL)*

*{*

*fclose(fp);*

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

*exit(1);*

*}*

*fread (&fileHeaderInfo,FILE\_HEADER\_SIZE, 1, fp);*

*fwrite(&fileHeaderInfo,FILE\_HEADER\_SIZE, 1, tmpFp);*

*printf("\n\tEnter Book ID NO. for delete:");*

*scanf("%d",&bookDelete);*

*while (fread (&addBookInfoInDataBase, sizeof(addBookInfoInDataBase), 1, fp))*

*{*

*if(addBookInfoInDataBase.books\_id != bookDelete)*

*{*

*fwrite(&addBookInfoInDataBase,sizeof(addBookInfoInDataBase), 1, tmpFp);*

*}*

*else*

*{*

*found = 1;*

*}*

*}*

*(found)? printf("\n\tRecord deleted successfully....."):printf("\n\tRecord not found");*

*getch();*

*fclose(fp);*

*fclose(tmpFp);*

*remove(FILE\_NAME);*

*rename("tmp.bin",FILE\_NAME);*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//function to update password*

*//---------------------------------------------------------------------------------------------------------*

*void updateCredential(void)*

*{*

*sFileHeader fileHeaderInfo = {0};*

*FILE \*fp = NULL;*

*unsigned char userName[MAX\_SIZE\_USER\_NAME] = {0};*

*unsigned char password[MAX\_SIZE\_PASSWORD] = {0};*

*headMessage("Update Credential");*

*fp = fopen(FILE\_NAME,"r+");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*fread (&fileHeaderInfo,FILE\_HEADER\_SIZE, 1, fp);*

*if (fseek(fp,0,SEEK\_SET) != 0)*

*{*

*fclose(fp);*

*printf("\n\tFacing issue while updating password\n");*

*exit(1);*

*}*

*printf("\n\n\t\t\tNew Username:");*

*fflush(stdin);*

*fgets(userName,MAX\_SIZE\_USER\_NAME,stdin);*

*printf("\n\n\t\t\tNew Password:");*

*fflush(stdin);*

*fgets(password,MAX\_SIZE\_PASSWORD,stdin);*

*strncpy(fileHeaderInfo.username,userName,sizeof(userName));*

*strncpy(fileHeaderInfo.password,password,sizeof(password));*

*fwrite(&fileHeaderInfo,FILE\_HEADER\_SIZE, 1, fp);*

*fclose(fp);*

*printf("\n\t\t\tYour Password has been changed successfully");*

*printf("\n\t\t\tLogin Again from starting ");*

*fflush(stdin);*

*getchar();*

*exit(1);*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//menu1 function of the program*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*void menuagain()*

*{*

*int choice = 0;*

*do*

*{*

*headMessage("MAIN MENU");*

*printf("\n\n\n\t\t\t1.Book Issue");*

*printf("\n\t\t\t2.Book Deposit");*

*printf("\n\t\t\t3.Administrator Menu");*

*printf("\n\t\t\t4.Update Password");*

*printf("\n\t\t\t0.Exit");*

*printf("\n\n\n\t\t\tEnter choice => ");*

*scanf("%d",&choice);*

*switch(choice)*

*{*

*case 1:*

*bookissue();*

*break;*

*case 2:*

*bookdeposit();*

*break;*

*case 3:*

*administratormenu();*

*break;*

*case 4:*

*updateCredential();*

*break;*

*case 0:*

*printf("\n\n\n\t\t\t\tThank you!!!\n\n\n\n\n");*

*exit(1);*

*break;*

*default:*

*printf("\n\n\n\t\t\tINVALID INPUT!!! Try again...");*

*}*

*//Switch Ended*

*}*

*while(choice!=0);*

*//Loop Ended*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//administrator menu function of the program*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*void administratormenu()*

*{*

*int choice = 0;*

*do*

*{*

*headMessage("ADMINISTRATOR MENU");*

*printf("\n\n\n\t\t\t1.Add Books");*

*printf("\n\t\t\t2.Search Books");*

*printf("\n\t\t\t3.View Books");*

*printf("\n\t\t\t4.Delete Book");*

*printf("\n\t\t\t5.Book Issue");*

*printf("\n\t\t\t6.Book Deposit");*

*printf("\n\t\t\t7.Update Password");*

*printf("\n\t\t\t8.Back");*

*printf("\n\t\t\t0.Exit");*

*printf("\n\n\n\t\t\tEnter choice => ");*

*scanf("%d",&choice);*

*switch(choice)*

*{*

*case 1:*

*addBookInDataBase();*

*break;*

*case 2:*

*searchBooks();*

*break;*

*case 3:*

*viewBooks();*

*break;*

*case 4:*

*deleteBooks();*

*break;*

*case 5:*

*bookissue();*

*break;*

*case 6:*

*bookdeposit();*

*break;*

*case 7:*

*updateCredential();*

*break;*

*case 8:*

*menuagain();*

*break;*

*case 0:*

*printf("\n\n\n\t\t\t\tThank you!!!\n\n\n\n\n");*

*exit(1);*

*break;*

*default:*

*printf("\n\n\n\t\t\tINVALID INPUT!!! Try again...");*

*}*

*//Switch Ended*

*}*

*while(choice!=0);*

*//Loop Ended*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//menu function of the program*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*void menu()*

*{*

*int choice = 0;*

*do*

*{*

*headMessage("MAIN MENU");*

*printf("\n\n\n\t\t\t1.Book Issue");*

*printf("\n\t\t\t2.Book Deposit");*

*printf("\n\t\t\t3.Administrator Menu");*

*printf("\n\t\t\t4.Update Password");*

*printf("\n\t\t\t0.Exit");*

*printf("\n\n\n\t\t\tEnter choice => ");*

*scanf("%d",&choice);*

*switch(choice)*

*{*

*case 1:*

*bookissue();*

*break;*

*case 2:*

*bookdeposit();*

*break;*

*case 3:*

*administratormenu();*

*break;*

*case 4:*

*updateCredential();*

*break;*

*case 0:*

*printf("\n\n\n\t\t\t\tThank you!!!\n\n\n\n\n");*

*exit(1);*

*break;*

*default:*

*printf("\n\n\n\t\t\tINVALID INPUT!!! Try again...");*

*}*

*//Switch Ended*

*}*

*while(choice!=0);*

*//Loop Ended*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//login password*

*//---------------------------------------------------------------------------------------------------------*

*void login()*

*{*

*unsigned char userName[MAX\_SIZE\_USER\_NAME] = {0};*

*unsigned char password[MAX\_SIZE\_PASSWORD] = {0};*

*int L=0;*

*sFileHeader fileHeaderInfo = {0};*

*FILE \*fp = NULL;*

*headMessage("Login");*

*fp = fopen(FILE\_NAME,"r");*

*if(fp == NULL)*

*{*

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

*exit(1);*

*}*

*fread (&fileHeaderInfo,FILE\_HEADER\_SIZE, 1, fp);*

*fclose(fp);*

*do*

*{*

*int ch=0;*

*printf("\n\n\n\t\t\t\tUsername:");*

*fgets(userName,MAX\_SIZE\_USER\_NAME,stdin);*

*printf("\n\t\t\t\tPassword:");*

*fgets(password,MAX\_SIZE\_PASSWORD,stdin);*

*if((!strcmp(userName,fileHeaderInfo.username)) && (!strcmp(password,fileHeaderInfo.password)))*

*{*

*menu();*

*}*

*else*

*{*

*printf("\n\t\t\t\tLogin Failed Enter Again Username & Password\n\n");*

*printf("\n\t\t\t\t1. For Try again\n\n");*

*printf("\n\t\t\t\t2. Forgot Username and Password\n\n");*

*printf("\t\t\t Enter your choice => ");*

*scanf("%d",&ch);*

*fflush(stdin);*

*if(ch==2)*

*{*

*updateCredential();*

*}*

*fflush(stdin);*

*L++;*

*}*

*}*

*while(L<=3);*

*if(L>3)*

*{*

*headMessage("Login Failed");*

*printf("\t\t\t\tSorry,Unknown User.");*

*getch();*

*system("cls");*

*}*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//function to check file exist or not*

*//---------------------------------------------------------------------------------------------------------*

*int isFileExists(const char \*path)*

*{*

*// Try to open file*

*FILE \*fp = fopen(path, "r");*

*int status = 0;*

*// If file does not exists*

*if (fp != NULL)*

*{*

*status = 1;*

*// File exists hence close file*

*fclose(fp);*

*}*

*return status;*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//to change password*

*//---------------------------------------------------------------------------------------------------------*

*void defpassword()*

*{*

*FILE \*fp = NULL;*

*int status = 0;*

*const char defaultUsername[] ="library\n";*

*const char defaultPassword[] ="library\n";*

*sFileHeader fileHeaderInfo = {0};*

*status = isFileExists(FILE\_NAME);*

*if(!status)*

*{*

*//create the binary file*

*fp = fopen(FILE\_NAME,"w");*

*if(fp != NULL)*

*{*

*//Copy default password*

*strncpy(fileHeaderInfo.password,defaultPassword,sizeof(defaultPassword));*

*strncpy(fileHeaderInfo.username,defaultUsername,sizeof(defaultUsername));*

*fwrite(&fileHeaderInfo,FILE\_HEADER\_SIZE, 1, fp);*

*fclose(fp);*

*}*

*}*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------START OF THE PROGRAM----------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

*int main()*

*{*

*defpassword();*

*welcomeMessage();*

*login();*

*return 0;*

*}*

*//---------------------------------------------------------------------------------------------------------*

*//---------------------------------------END OF THE PROGRAM------------------------------------------------*

*//---------------------------------------------------------------------------------------------------------*

***Conclusion and future scope***

*This C Program provides a computerized version of Library Management System which will benefit the students as well as the staff of the library. This Library Management System Project in C can perform functions such as add books, return books, issue books, delete record of books issued, view record of books issued, search for books, and more. File handling has been extensively used in this project for almost all functions. It also has facility for student login where students can login and see status of books issued as well as request for books. Scope:*

*The different areas where this program can be used are :*

* *Any educational institute can make use of it for maintaining the records of books issued by students and providing information of the books available at the library.*
* *It can be used in offices and modifications can be easily done according to requirements.*
* *There is another future scope of this facility that more features like money transactions for issuing books can be added.*

***references***

* *Kanetkar Yashavant, Let Us C, BPB Publication*
* *Balagurusamy E, Programming In ANSI C, the Tata McGraw-Hill Companies*
* [*https://www.geeksforgeeks.org/e-library-management-system/*](https://www.geeksforgeeks.org/e-library-management-system/)
* [*www.google.com*](http://www.google.com)
* [*www.cprogramming.com*](http://www.cprogramming.com)

***Thanking you***

***In anticipation***

***By…***

Himanshu dixit*(pdf creator)*

ARYAN RAZDAN

Ashutosh pradhan

Aryaman vishnoi