PROJECT OF OBJECT ORIENTED PROGRAMMING USING C ++

The Library Management System

The Library Management System automates the basic library functions to aid in the day-to-day operations of a library. The purpose of the system is to allow for storing details of a large number of books and allow for add, borrow, return facilities separately to staff and students. Different privileges are given to different types of users.

It also maintains data about books about books, teachers, students records that are required during various library operations. The software aims to make the system user friendly and efficient. Some of the features involved are date of issuing the book and return the book. If the there is any delay in returning the book fine should be paid

Modules of Library Management System

Add Book

Add Member

Modify Book

Modify Member

Delete Book

Delete Member

Issue Book

Return Book

Functions of Library Management System

The functions that the Library Management System provides are as follows:

1. INSERT: This operation is performed when new data needs to be added to the system, for e.g. when department purchases a new book, the book's entry is inserted in the books database. This option has three choices:

a) Book: This choise allows entering data about newly purchased books into the books database. The data entered includes book's author, title, publisher, cost and various other fields provided in the form. The data must be accurate and must be entered in the correct format as indicated in the forms.

b) Teacher: This option is used for entering data for a new teacher in the teacher's database. This option is chosen when a teacher joins college.

c) Student: This will enter new record for a student in student's database. This option is chosen when a student is enrolled in the college.

2. DELETE: This operation clears the existing records in the various databases. It is used when for e.g. a member leaves college or when book is disposed of from library. But care mustbe taken while performing this operation and permissiontaken from the head of library because the system could loose any important data.

It can be performed on all databases and on three choices are:

a) Book: This will enter a null value for the book whose accession number is entered in the fieldprovided in the respected form. This operation is done when a book is disposed of the library.

b) Teacher: This will clear the record for a particular teacher whose identification number is entered. This option is chosen when a teacher leaves college.

c) Student: This will clear the record for the particular student whose record needs to be deleted by entering her roll number in the required field. This option is chosen when a student leaves college.

3. UPDATE: This function updates data in the various records. This operation is supported by all the three enteries:

a) Book: This function generally would not be required for updating a book's status as that data wouldn't change.

b) Teacher: This will update the data of particular teacher, whose data has changed like address, phone number, etc. by entering her identification number.

3. UPDATE: This function updates data in the various records. This operation is supported by all the three enteries:

a) Book: This function generally would not be required for updating a book's status as that data wouldn't change.

b) Teacher: This will update the data of particular teacher, whose data has changed like address, phone number, etc. by entering her identification number.

c) Student: This will update the data of student like address, course, etc. by entering student's roll number.

4. SEARCH: This function is used to search particular data from the database. This function can search for data related to all the three entities:

a) Book: To search for a particular book, to know whether it is currently available in library or not. This can be done by entering value in any one or more fields in the form to perform the search such as title or author name.

b) Teacher: This will find out the particular teacher who currently has the book for which search is being carried out.

c) Student: This will find out the particular student who possesses the particular book.

5. ISSUE: This operation is used for issuing a book to a member of the library. For this operation to be successful the member must meet some criteria like she shouldnot have issued books to her maximum limit previously. All these checks are done by software. If the operation is successful, then the system automatically stores the date of issue and the due date by which the book must be returned.

a) Student: When a student loans a book, the entry of the book is stored automatically in the student's database with the due date of that book.

b) Teacher: In case a member of teaching staff loans a book the entry is stored in teacher's database with the due date of the book.

3. UPDATE: This function updates data in the various records. This operation is supported by all the three enteries:

a) Book: This function generally would not be required for updating a book's status as that data wouldn't change.

b) Teacher: This will update the data of particular teacher, whose data has changed like address, phone number, etc. by entering her identification number.

c) Student: This will update the data of student like address, course, etc. by entering student's roll number.

4. SEARCH: This function is used to search particular data from the database. This function can search for data related to all the three entities:

a) Book: To search for a particular book, to know whether it is currently available in library or not. This can be done by entering value in any one or more fields in the form to perform the search such as title or author name.

b) Teacher: This will find out the particular teacher who currently has the book for which search is being carried out.

c) Student: This will find out the particular student who possesses the particular book.

5. ISSUE: This operation is used for issuing a book to a member of the library. For this operation to be successful the member must meet some criteria like she shouldnot have issued books to her maximum limit previously. All these checks are done by software. If the operation is successful, then the system automatically stores the date of issue and the due date by which the book must be returned.

a) Student: When a student loans a book, the entry of the book is stored automatically in the student's database with the due date of that book.

b) Teacher: In case a member of teaching staff loans a book the entry is stored in teacher's database with the due date of the book.

6. RETURN: Using this operation a member returns the items, which she loaned, from the library back to it. If the book, which is loaned is not returned within specified time the member ends up as a defaulter and she is required to pay fine which is calculated automatically by the software.

a) Teacher: It will delete the corresponding entry made in teacher's database.

b) Student: It will delete the corresponding entry made in student's database.

7. DISPLAY: This is used to display each and every record, i.e. record of every book, teacher and student in the library.

a) Book: Record of every book, i.e. it's accession number, author name, publisher name, etc.

b) Teacher: Record of every teacher, i.e. her id, department, no. of books issued, etc., who is member of the college library.

c) Student: Record of every student, i.e. her roll number, course, no of books issued, etc., who is member of the college library.

6. RETURN: Using this operation a member returns the items, which she loaned, from the library back to it. If the book, which is loaned is not returned within specified time the member ends up as a defaulter and she is required to pay fine which is calculated automatically by the software.

a) Teacher: It will delete the corresponding entry made in teacher's database.

b) Student: It will delete the corresponding entry made in student's database.

7. DISPLAY: This is used to display each and every record, i.e. record of every book, teacher and student in the library.

a) Book: Record of every book, i.e. it's accession number, author name, publisher name, etc.

b) Teacher: Record of every teacher, i.e. her id, department, no. of books issued, etc., who is member of the college library.

c) Student: Record of every student, i.e. her roll number, course, no of books issued, etc., who is member of the college library.

8. EXIT: This takes user out of the application.This will update the data of student like address, course, etc. by entering student's roll number.

SOURCE CODE IS HERE……

```
#include<iostream>

#include<conio.h>

#include<string.h>

#include<process.h>

using namespace std;

class books

 {

 public:

 int stock;

 char author[20], publisher[20];

 char bookname[20];
```

```
 float price;
 void loadbooks();
 void display();
};
void books::loadbooks()
{
 cout<<"\nEnter Book Name:";
 cin>>bookname;
 cout<<"\nEnter Author Name:";
 cin>>author;
 cout<<"\nEnter Publisher Name:";
 cin>>publisher;
 cout<<"\nEnter Price:";
 cin>>price;
 cout<<"\nEnter Stock:";
 cin>>stock;
 cout<<"\n----------\n";
}
void books::display()
 {
 cout<<"\nName of the Book:"<<bookname;
 cout<<"\nAuthor of the Book:"<<author;
 cout<<"\nPublisher of the Book:"<<publisher;
 cout<<"\nPrice of the Book:"<<price;
 cout<<"\nStock Present:"<<stock;
```

```
 cout<<"\n----------\n";
}
int main()
{
 books ob[10];
 int ch, n;
 do
{
  cout<<"\n****\n";
  cout<<"\n1.Load Books\n2.Display\n3.Search\n4.Exit\n";
  cout<<"\n\nEnter your Choice:";
  cin>>ch;
  switch(ch)
  {
  case 1: cout<<"Enter Number of Books:";
   cin>>n;
   for(int i=0;i<n;i++)
   ob[i].loadbooks();
   break;
  case 2:
   for(int i=0;i<n;i++)
   ob[i].display();
   break;
  case 3:
char bname[20], aname[20];
```

```
 cout<<"Enter name of the Book:";
 cin>>bname;
 cout<<"Enter name of the Author:";
 cin>>aname;
 for(int i=0;i<n;i++)
 {
  if(strcmp(bname, ob[i].bookname)==0&&strcmp(aname,ob[i].author))
  {
   cout<<"\nBook Present\n\n";
   cout<<"\nName of the Book:"<<ob[i].bookname;
   cout<<"\nAuthor of the Book:"<<ob[i].author;
   cout<<"\nPublisher of the Book:"<<ob[i].publisher;
   cout<<"\nPrice of the Book:"<<ob[i].price;
   cout<<"\nStock Present:"<<ob[i].stock;
   cout<<"\n----------\n";
   break;
  }
  else
  {
   cout<<"Not Present!!";
   break;
  }
 }
 break;
default: cout<<"Enter a valid choice!!";
```

```
  case 4: exit(1);
  }
 }while(1);
}
```

OUTPUT IS GIVEN BELOW---------

```
**********************************************
 WELCOME TO THE LIBRARY MANAGEMENT SYSTEM
**********************************************
1.Add Books
2.Display
3.Search
4.Exit
Enter your Choice:1
Enter Number of Books:2
Enter Book Name:pc
Enter Author Name:ak
Enter Publisher Name:gold
Enter Price:234
Enter Stock:1
----------
Enter Book Name:se
Enter Author Name:ag
Enter Publisher Name:bold
Enter Price:342
Enter Stock:2
```

****

WELCOME TO THE LIBRARY MANAGEMENT SYSTEM

****

1.Add Books

2.Display

3.Search

4.Exit

Enter your Choice:3

Enter name of the Book:sety

Enter name of the Author:ag

Not Present!!**********

WELCOME TO THE LIBRARY MANAGEMENT SYSTEM

****

1.Add Books

2.Display

3.Search

4.Exit

Enter your Choice:2

Name of the Book:pc

Author of the Book:ak

Publisher of the Book:gold

Price of the Book:234

Stock Present:1

Name of the Book:se

Author of the Book:ag

Publisher of the Book:bold

Price of the Book:342

Stock Present:2

****

WELCOME TO THE LIBRARY MANAGEMENT SYSTEM

****

1.Add Books

2.Display

3.Search

4.Exit

Enter your Choice:4

compilation terminated.....