Data Base Lab1(Mini Project)

Code:

#include<iostream>

#include<fstream>

#include<iomanip>

#ifdef \_WIN32

#include <Windows.h>

#else

#include <unistd.h>

#endif

#include <cstdlib>

using namespace std;

class library\_management\_system {

string Title;

string author;

int ISBN, Old\_IS, Old\_T, Old\_A;

int Number\_of\_copy, Old\_NC;

int count;

string\* A;

string\* T;

int\* IS;

int\* No;

fstream Author;

fstream title;

fstream Isbn;

fstream Number\_Of\_Copy;

public:

library\_management\_system()

{

count = 0;

Old\_T = -1;

Old\_A = -1;

Old\_IS = -1;

Old\_NC = -1;

Title = " ";

author = " ";

ISBN = -1;

Number\_of\_copy = -1;

}

int Count()

{

count = 0;

Author.open("Auhtor\_name.txt", ios::in);

while (Author >> author)

{

++count;

}

Author.close();

return count;

}

void create()

{

cout << " Enter the title of book :";

cin >> Title;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Title;

cout << " Not a valid input \n";

cout << " Input must be string\n";

}

cout << " Enter the name of author :";

cin >> author;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> author;

cout << " Not a valid input \n";

cout << " Input must be string \n";

}

cout << " Enter the ISBN number :";

cin >> ISBN;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> ISBN;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

cout << " Enter the number of copies :";

cin >> Number\_of\_copy;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Number\_of\_copy;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

Author.open("Auhtor\_name.txt", ios::app);

title.open("Book\_Title.txt", ios::app);

Isbn.open("ISBN\_Number.txt", ios::app);

Number\_Of\_Copy.open("Number\_of\_copies.txt", ios::app);

Author << author << endl;

title << Title << endl;

Isbn << ISBN << endl;

Number\_Of\_Copy << Number\_of\_copy << endl;

Author.close();

title.close();

Isbn.close();

Number\_Of\_Copy.close();

}

void update(int n)

{

A = new string[n];

T = new string[n];

IS = new int[n];

No = new int[n];

int choice;

Old\_T = -1;

Old\_A = -1;

Old\_IS = -1;

Old\_NC = -1;

Author.open("Auhtor\_name.txt", ios::in);

title.open("Book\_Title.txt", ios::in);

Isbn.open("ISBN\_Number.txt", ios::in);

Number\_Of\_Copy.open("Number\_of\_copies.txt", ios::in);

do {

system("cls");

cout << "\n \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout << " | Update Any Book | \n";

cout << " |\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\n\n";

cout << " Press 1 for Auhtor name Updation \n";

cout << " Press 2 for book Title Updation \n";

cout << " Press 3 for ISBN number Updation \n";

cout << " Press 4 for Number of copies Updation \n";

cout << " Press 5 for Exit \n";

cin >> choice;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> choice;

cout << " Not a valid input \n";

cout << " Input must be integer between(1 to 5)\n";

}

switch (choice) {

case 1: {

cout << "Enter the old ISBN number of book :";

cin >> Old\_A;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Old\_A;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

cout << "Enter the Author name :";

cin >> author;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> author;

cout << " Not a valid input \n";

cout << " Input must be string\n";

}

break;

}

case 2: {

cout << "Enter the old ISBN number of book :";

cin >> Old\_T;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Old\_T;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

cout << "Enter the new title :";

cin >> Title;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Title;

cout << " Not a valid input \n";

cout << " Input must be string \n";

}

break;

}

case 3: {

cout << "Enter the old ISBN number :";

cin >> Old\_IS;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Old\_IS;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

cout << "Enter the new ISBN number :";

cin >> ISBN;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> ISBN;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

break;

}

case 4: {

cout << "Enter the ISBN number of book :";

cin >> Old\_NC;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Old\_NC;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

cout << "Enter the new Number of copies :";

cin >> Number\_of\_copy;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Number\_of\_copy;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

break;

}

case 5: {

cout << " Terminate Update function \n";

break;

}

}

} while (choice != 5);

bool exist = true;

for (int i = 0; i < n; ++i)

{

Author >> A[i];

title >> T[i];

Isbn >> IS[i];

Number\_Of\_Copy >> No[i];

if (Old\_A != -1 && Old\_A == IS[i])

{

A[i] = author;

exist = false;

}

else if (Old\_IS != -1 && Old\_IS == IS[i])

{

IS[i] = ISBN;

exist = false;

}

else if (Old\_T != -1 && Old\_T == IS[i])

{

T[i] = Title;

exist = false;

}

else if (Old\_NC != -1 && Old\_NC == IS[i])

{

No[i] = Number\_of\_copy;

exist = false;

}

}

Author.close();

title.close();

Isbn.close();

Number\_Of\_Copy.close();

Author.open("Auhtor\_name.txt", ios::out);

title.open("Book\_Title.txt", ios::out);

Isbn.open("ISBN\_Number.txt", ios::out);

Number\_Of\_Copy.open("Number\_of\_copies.txt", ios::out);

for (int i = 0; i < n; ++i)

{

Author << A[i] << endl;

title << T[i] << endl;

Isbn << IS[i] << endl;

Number\_Of\_Copy << No[i] << endl;

}

Author.close();

title.close();

Isbn.close();

Number\_Of\_Copy.close();

delete[] A;

delete[] T;

delete[]IS;

delete[]No;

if (exist)

{

cout << " This data dose not exist \n";

}

else {

cout << " Updation done \n";

}

}

void deletion(int n)

{

A = new string[n];

T = new string[n];

IS = new int[n];

No = new int[n];

int choice;

bool exist = true;

Author.open("Auhtor\_name.txt", ios::in);

title.open("Book\_Title.txt", ios::in);

Isbn.open("ISBN\_Number.txt", ios::in);

Number\_Of\_Copy.open("Number\_of\_copies.txt", ios::in);

for (int i = 0; i < n; ++i)

{

Author >> A[i];

title >> T[i];

Isbn >> IS[i];

Number\_Of\_Copy >> No[i];

}

cout << "Enter the ISBN number of book :";

cin >> ISBN;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> Old\_NC;

cout << " Not a valid input \n";

cout << " Input must be integer \n";

}

Author.close();

title.close();

Isbn.close();

Number\_Of\_Copy.close();

Author.open("Auhtor\_name.txt", ios::out);

title.open("Book\_Title.txt", ios::out);

Isbn.open("ISBN\_Number.txt", ios::out);

Number\_Of\_Copy.open("Number\_of\_copies.txt", ios::out);

for (int i = 0; i < n; ++i)

{

if (ISBN != IS[i])

{

Author << A[i] << endl;

title << T[i] << endl;

Isbn << IS[i] << endl;

Number\_Of\_Copy << No[i] << endl;

}

else {

exist = false;

}

}

if (exist)

{

cout << " This data dose not exist \n";

}

else {

cout << " Deletion done \n";

}

Author.close();

title.close();

Isbn.close();

Number\_Of\_Copy.close();

delete[] A;

delete[] T;

delete[]IS;

delete[]No;

}

void display(int n)

{

system("cls");

cout << "\n Library Managment System \n\n";

if (n == 0)

{

cout << "Library is Empty \n";

return;

}

A = new string[n];

T = new string[n];

IS = new int[n];

No = new int[n];

Author.open("Auhtor\_name.txt", ios::in);

title.open("Book\_Title.txt", ios::in);

Isbn.open("ISBN\_Number.txt", ios::in);

Number\_Of\_Copy.open("Number\_of\_copies.txt", ios::in);

for (int i = 0; i < n; ++i)

{

Author >> A[i];

title >> T[i];

Isbn >> IS[i];

Number\_Of\_Copy >> No[i];

}

Author.close();

title.close();

Isbn.close();

Number\_Of\_Copy.close();

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout << "| Sr " << "| Title of book " << "| Author\_Name " << "| ISBN number " << "| Copies |" << endl;

for (int i = 0; i < n; ++i)

{

cout << "|\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_|\n";

cout << "| " << i + 1 << " |" << setw(15) << T[i] << " |" << setw(15) << A[i] << " |" << setw(10) << IS[i] << " |" << setw(6) << No[i] << " |" << endl;

}

cout << "|\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_|\n";

}

};

int main()

{

system("color 9E");

library\_management\_system obj;

int choice;

do {

cout << "\n \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout << " | Welcome To Library Management | \n";

cout << " |\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\n\n";

cout << "Add a new book (press 1)\n";

cout << "Update any book (press 2)\n";

cout << "Delete any book (press 3)\n";

cout << "Display all books (press 4)\n";

cout << "Exit (press 5)\n";

cin >> choice;

while (cin.fail()) {

cin.clear();

cin.ignore();

cin >> choice;

cout << " Not a valid input \n";

cout << " Input must be integer between(1 to 5)\n";

}

switch (choice) {

case 1:

{

system("cls");

cout << "\n \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout << " | ADD A NEW BOOK | \n";

cout << " |\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\n\n";

obj.create();

break;

}

case 2:

{

obj.update(obj.Count());

break;

}

case 3:

{system("cls");

cout << "\n \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout << " | Delete Any Book | \n";

cout << " |\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\n\n";

obj.deletion(obj.Count());

break;

}

case 4:

{

obj.display(obj.Count());

break;

}

case 5:

{

cout << " Program Terminated \n";

cout << " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n\n";

}

default: {

cout << " Invalid Input \n";

}

}

} while (choice != 5);

system("pause");

}

Screenshot:









