C++ Programming

**CD Management System**



***Efforts By : OJASSVI PRADHAN***

***XII-B***

***Carmel Convent School***

***ACKNOWLEDGEMENT***

***This project is made by OJASSVI PRADHAN and would like to express my sincere gratitude to my computer science teacher Tr. Suchanda Sen for her vital support, guidance and encouragement – without which this project would not have come forth.***

***(Ojassvi Pradhan)***

**CERTIFICATE**

This is to certify that **Ojassvi Pradhan** of class **XII-B** has completed the project **“CD Management System”** under my guidance and supervision in the academic year **2014-15**.

*(Tr. Suchanda Sen)*

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **No.** | **Title** |  |
| 1. | Objective of the Project |  |
| 2. | Working of the Program |  |
| 3. | Structure Charts |  |
|  | * Class CD |  |
|  | * void main( ) |  |
| 4. | Program Flow Charts |  |
| 5. | Classes and Functions Used |  |
| 6. | Program Coding |  |
| 7. | Sample Outputs |  |
| 8. | Bibliography |  |

**OBJECTIVE**

This software can be used to handle the inventory management in a CD store.

* It keeps record of all the CDs present in the CDs Store.
* We can add records for the fresh arrivals.
* We can search the CDs available in the store, by various modes like singer name, CD name, release, and cost.
* It has a unique feature of BACKUP to save all our records and also, if anything goes wrong, then we can restore our last backup file.

WORKING

*This C++ project is based on CD Management System. This program uses the concept of object-oriented programming and data file handling. Database is a collection of interrelated data to serve multiple applications. That is database programs create files of information. So we see that files are worked with most, inside the program.*

*The program helps us to perform several tasks that have been explained in detail as below:*

1. *Adding the details of a New CD:*

*To perform this task the program uses the technique of opening the existing “CD.dat” file for reading and a “temp.dat” file for writing. The existing records are read from the “CD.dat” file and are written in the “temp.dat” file. The new record is added in the “temp.dat” file and both the files are closed. The “CD.dat” file is then removed and the “temp.dat” file is renamed to “CD.dat”. Thus a new record is added at the end of the “CD.dat” file.*

1. *Modifying the data of an CD:*

*To perform this task the program uses the technique of opening the existing “CD.dat” file for reading and a “temp.dat” file for writing. The CD number to be modified is inputted from the user. All the records are read from the “CD.dat” file and each record’s employee number is compared with the one supplied by the user. Every record, after reading, is written into the “temp.dat” file except when the CD number match is successful. When the match is successful the user is asked to enter the new data. Then this data is added to the “temp.dat” file. Both the files are then closed. The “CD.dat” file is then removed and the “temp.dat” file is renamed to “CD.dat”. Thus the required CD record is modified from the “CD.dat” file.*

1. *Deleting a CD’s Details:*

*To perform this task the program uses the technique of opening the existing “CD.dat” file for reading and a “temp.dat” file for writing. The CD number to be deleted is inputted from the user. The existing records are read from the “CD.dat” file and each record’s CD number is compared with the one supplied by the user. Every record, after reading, is written into the “temp.dat” file except when the CD number match is successful. Thus all the records except for the one to be deleted are copied into the “temp.dat” file. Both the files are then closed. The “CD.dat” file is then removed and the “temp.dat” file is renamed to “CD.dat”. Thus the required CD record is deleted from the “CD.dat” file.*

1. *Searching for a CD:*

*To perform this task the program uses the technique of opening the existing “CD.dat” file for reading. The detail to be searched is inputted from the user. The CD records are then read one by one from the “CD.dat” file and each record’s CD detail is compared with the one supplied by the user. On finding a match, the program displays the details of that particular CD. Otherwise it displays that the reqd. CD’s details does not exists. The files are then closed. Thus a particular CD is searched for by the program.*

1. *Displaying the list of all CDs:*

*To perform this task the program uses the technique of opening the existing “CD.dat” file for reading. The CD records are then read one by one from the “CD.dat” file and only the relevant details are displayed. The file is then closed. Thus the list of all CDs is displayed.*

STRUCTURE CHARTS

**Class CD**

-The main CD class. Used for inputting and displaying data.

**accept()**

-To input values.

**display()**

-To output values.

**Read()**

- To read the data.

**rbno()**

-To return CD no.

**Search()**

-To search records.

**\*rbname()**

-To return CD name.

**shtype()**

-To search by type of CD.

**searchbname()**

-To search by CD name.

**shcost()**

-To search by cost.

**shaname()**

-To search by singer’s name.

**shrelease()**

-To search by year of release.

**Class CD**

-The main CD class. Used for inputting and displaying data.

**write()**

-To write records in CD.dat.

**update()**

-To update values.

**mainmenu()**

-To display main menu.

**menu()**

-To display search menu.

**restore()**

-To restore all records.

**backup()**

-To create a backup of all records.

**del()**

-To delete records.

**FLOWCHART**

**START**

ENTER PASSWORD

If (pwd = Value)

Choose the options

1. Add a Record

2. Display All the

Records

3. Modify a Record

4. Search a Record

5. Delete a Record

6. Create a Backup

7. Restore the Last

Backup File

8. Exit

Print “Invalid Password”

If Option = 8

Go To Particular Option

Go To Previous Menu

EXIT

Choose the Search Option

1 Search by CD Number

2. Search by CD Name

3. Search by Singer Name

4. Search by Release

5. Search by Type

6. Search by Cost

7. Back to Previous Menu

8. Exit

If Option = 4

Go To Particular Option

Go To Previous Menu

EXIT

If Option = 8

Print “Thank You

Have a Nice Day”

STOP

Print “Thank You

Have a Nice Day”

**Classes and Functions Used**

* **CLASSES:**
* CD
  + **MEMBER FUNCTIONS OF CLASS:**
* crno();
* accept();
* accept1();
* read();
* search();
* searchcname();
* shsname();
* shtype();
* shcost();
* shrelease();
  + **GLOBAL MEMBER FUNCTIONS OF CLASS:**
* write();
* update();
* del();
* backup();
* restore();
* menu();
* mainmenu();
* **DATA FILES USED**
* CD.dat (Original File Containing all data)
* temp.dat (A Temporary File for keeping deleted data)

HEADER FILES

1.fstream.h

2.conio.h

3.stdio.h

4.string.h

5.dos.h

6.process.h

Coding

#include<fstream.h>

#include<conio.h>

#include<stdio.h>

#include<string.h>

#include<process.h>

#include<dos.h>

void mainmenu();

class CD

{

private:

int cno;

char cname[30];

char sname[30];

char type[10];

int nocopy;

char yr[5];

float cost;

public:

CD()

{

cno=0;

nocopy=0;

cost=0;

}

int rcno() //for asscessing private member outside class

{

return cno;

}

void accept()

{

clrscr();

int count=0;

fstream f1;

f1.open("CD.dat",ios::in|ios::binary);

while(f1.read((char\*)this,sizeof(CD)))

count++;

if(count==0)

cno=1;

else

cno++;

f1.close();

cout<<"\nEnter The CD Number:- "<<cno;

cout<<"\nEnter The CD Title:- ";

gets(cname);

cout<<"\nEnter Singer's Name:- ";

gets(sname);

cout<<"\nEnter The Type of The CD:- ";

gets(type);

cout<<"\nEnter No. of Copies:- ";

cin>>nocopy;

cout<<"\nEnter Year of Release:- ";

cin>>yr;

cout<<"\nEnter the Cost of CD:- ";

cin>>cost;

}

void accept1()

{

cout<<"\nEnter The CD Title:- ";

gets(cname);

cout<<"\nEnter Singer's Name:- ";

gets(sname);

cout<<"\nEnter The Type of The CD:- ";

gets(type);

cout<<"\nEnter No. of Copies:- ";

cin>>nocopy;

cout<<"\nEnter Year of Release:- ";

cin>>yr;

cout<<"\nEnter the Cost of The CD:- ";

cin>>cost;

}

void read();

void search();

void searchcname();

void shsname();

void shtype();

void shcost();

void shrelease();

};

void write()

{

fstream f1("CD.dat",ios::app|ios::binary);

CD e;

e.accept();

f1.write((char\*)&e,sizeof(e));

cout<<"\nRECORD ADDED SUCCESSFULLY!!!!!!!!!!\n";

getch();

f1.close();

}

void CD::read()

{

clrscr() ;

fstream file ;

file.open("CD.dat",ios::in|ios::binary);

file.seekg(0) ;

int row = 5 , found = 0 , pageno = 1 ;

gotoxy(18,1) ;

cout<<"LIST OF CDS" ;

gotoxy(1,3) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

while (file.read((char \*) this, sizeof(CD)))

{

delay(50) ;

found = 1 ;

gotoxy(2,row) ;

cout<<cno;

gotoxy(14,row) ;

cout<<cname;

gotoxy(27,row) ;

cout<<sname;

gotoxy(42,row) ;

cout<<type;

gotoxy(55,row) ;

cout<<yr;

gotoxy(66,row) ;

cout<<nocopy;

gotoxy(74,row) ;

cout<<cost;

if (row==23)

{

row=5 ;

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

pageno++ ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

clrscr() ;

cout<<"LIST OF CDS" ;

gotoxy(1,5) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

}

else

row++ ;

}

if ( !found )

{

gotoxy(1,5) ;

cout<<"Records not found" ;

}

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

file.close () ;

}

void update()

{

clrscr();

fstream f1("CD.dat",ios::in|ios::out|ios::binary);

CD e;

int n,flag=0,record=0;;

cout<<"\nEnter CD No. to be modified\n";

cin>>n;

while(f1.read((char\*)&e,sizeof(e)))

{

record++;

if(e.rcno()==n)

{

cout<<"\nEnter New Values\n";

e.accept1();

f1.seekp((record-1)\*sizeof(e));

f1.write((char\*)&e,sizeof(e));

cout<<"\n\n\n\nRecord Modified Sucessfully\n\n\n";

getch();

flag=1;

break;

}

}

if(!flag)

cout<<"\nRecord Not Found!!!!!!!\n";delay(200);

f1.close();

}

void del()

{

clrscr();

fstream f1,f2;

f1.open("CD.dat",ios::in|ios::binary);

f2.open("temp.dat",ios::out|ios::binary);

CD e;

int n,flag=0;

cout<<"\nEnter CD Number to be Deleted\n";

cin>>n;

while(f1.read((char\*)&e,sizeof(e)))

{

if(e.rcno()!=n)

f2.write((char\*)&e,sizeof(e));

else

flag=1;

}

if(flag)

cout<<"\nRecord Deleted!!!!!!!\n";

else

cout<<"\nRecord Not Found!!!!!\n";

getch();

f1.close();

f2.close();

remove("CD.dat");

rename("temp.dat","CD.dat");

}

void CD::search()

{

clrscr() ;

fstream file ;

file.open("CD.dat",ios::in|ios::binary);

file.seekg(0) ;

int n;

cout<<"\n\nEnter the CD number to be searched\n";

cin>>n;

clrscr();

int row = 5 , found = 0 , pageno = 1 ;

gotoxy(18,1) ;

cout<<"LIST OF CDS" ;

gotoxy(1,3) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

while (file.read((char \*) this, sizeof(CD)))

{

if(cno==n)

{

delay(50) ;

found = 1 ;

gotoxy(2,row) ;

cout<<cno;

gotoxy(14,row) ;

cout<<cname;

gotoxy(27,row) ;

cout<<sname;

gotoxy(42,row) ;

cout<<type;

gotoxy(55,row) ;

cout<<yr;

gotoxy(66,row) ;

cout<<nocopy;

gotoxy(74,row) ;

cout<<cost;

if (row==23)

{

row=5;

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

pageno++ ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

clrscr() ;

cout<<"LIST OF CDS" ;

gotoxy(1,5) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

}

else

row++;

}

}

if ( !found )

{

gotoxy(1,5) ;

cout<<"Records not found" ;

}

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

file.close () ;

}

void CD::searchcname()

{

clrscr() ;

fstream file ;

file.open("CD.dat",ios::in|ios::binary);

file.seekg(0) ;

char n[20];

cout<<"\n\nEnter the CD name to be searched\n";

gets(n);

clrscr();

int row = 5 , found = 0 , pageno = 1 ;

gotoxy(18,1) ;

cout<<"LIST OF CDS" ;

gotoxy(1,3) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

while (file.read((char \*) this, sizeof(CD)))

{

if(strcmpi(cname,n)==0)

{

delay(50) ;

found = 1 ;

gotoxy(2,row) ;

cout<<cno;

gotoxy(14,row) ;

cout<<cname;

gotoxy(27,row) ;

cout<<sname;

gotoxy(42,row) ;

cout<<type;

gotoxy(55,row) ;

cout<<yr;

gotoxy(66,row) ;

cout<<nocopy;

gotoxy(74,row) ;

cout<<cost;

if (row==23)

{

row=5 ;

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

pageno++ ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

clrscr() ;

cout<<"LIST OF CDS" ;

gotoxy(1,5) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

}

else

row++ ;

}

}

if ( !found )

{

gotoxy(1,5) ;

cout<<"Records not found" ;

}

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

file.close () ;

}

void CD::shsname()

{

clrscr() ;

fstream file ;

file.open("CD.dat",ios::in|ios::binary);

file.seekg(0) ;

char n[20];

cout<<"Enter the singer's name whose CDs is to be searched\n";

gets(n);

clrscr();

int row = 5 , found = 0 , pageno = 1 ;

gotoxy(18,1) ;

cout<<"LIST OF CDS" ;

gotoxy(1,3) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

while (file.read((char \*) this, sizeof(CD)))

{

if(strcmpi(sname,n)==0)

{

delay(50) ;

found = 1 ;

gotoxy(2,row) ;

cout<<cno;

gotoxy(14,row) ;

cout<<cname;

gotoxy(27,row) ;

cout<<sname;

gotoxy(42,row) ;

cout<<type;

gotoxy(55,row) ;

cout<<yr;

gotoxy(66,row) ;

cout<<nocopy;

gotoxy(74,row) ;

cout<<cost;

if (row==23)

{

row=5 ;

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

pageno++ ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

clrscr() ;

cout<<"LIST OF CDS" ;

gotoxy(1,5) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

}

else

row++ ;

}

}

if ( !found )

{

gotoxy(1,5) ;

cout<<"Records not found" ;

}

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

file.close () ;

}

void CD::shtype()

{

clrscr() ;

fstream file ;

file.open("CD.dat",ios::in|ios::binary);

file.seekg(0) ;

char n[20];

cout<<"Enter the Type of CDs to searched\n";

gets(n);

clrscr();

int row = 5 , found = 0 , pageno = 1 ;

gotoxy(18,1) ;

cout<<"LIST OF CDS" ;

gotoxy(1,3) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

while (file.read((char \*) this, sizeof(CD)))

{

if(strcmpi(type,n)==0)

{

delay(50) ;

found = 1 ;

gotoxy(2,row) ;

cout<<cno;

gotoxy(14,row) ;

cout<<cname;

gotoxy(27,row) ;

cout<<sname;

gotoxy(42,row) ;

cout<<type;

gotoxy(55,row) ;

cout<<yr;

gotoxy(66,row) ;

cout<<nocopy;

gotoxy(74,row) ;

cout<<cost;

if ( row == 23 )

{

row=5 ;

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

pageno++ ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

clrscr() ;

cout<<"LIST OF CDS" ;

gotoxy(1,5) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

}

else

row++ ;

}

}

if ( !found )

{

gotoxy(1,5) ;

cout<<"Records not found" ;

}

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

file.close () ;

}

void CD::shrelease()

{

clrscr() ;

fstream file ;

file.open("CD.dat",ios::in|ios::binary);

file.seekg(0) ;

char n[20];

cout<<"Enter the Year of Release to be searched\n";

gets(n);

clrscr();

int row = 5 , found = 0 , pageno = 1 ;

gotoxy(18,1) ;

cout<<"LIST OF CDS" ;

gotoxy(1,3) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

while (file.read((char \*) this, sizeof(CD)))

{

if(strcmpi(yr,n)==0)

{

delay(50) ;

found = 1 ;

gotoxy(2,row) ;

cout<<cno;

gotoxy(14,row) ;

cout<<cname;

gotoxy(27,row) ;

cout<<sname;

gotoxy(42,row) ;

cout<<type;

gotoxy(55,row) ;

cout<<yr;

gotoxy(66,row) ;

cout<<nocopy;

gotoxy(74,row) ;

cout<<cost;

if (row==23)

{

row=5 ;

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

pageno++ ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

clrscr() ;

cout<<"LIST OF CDS" ;

gotoxy(1,5) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

}

else

row++ ;

}

}

if ( !found )

{

gotoxy(1,5) ;

cout<<"Records not found" ;

}

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

file.close () ;

}

void CD::shcost()

{

clrscr() ;

fstream file ;

file.open("CD.dat",ios::in|ios::binary);

file.seekg(0) ;

int n;

cout<<"Enter the minimum cost to see the CDs\n";

cin>>n;

clrscr();

int row = 5 , found = 0 , pageno = 1 ;

gotoxy(18,1) ;

cout<<"LIST OF CDS" ;

gotoxy(1,3) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

while (file.read((char \*) this, sizeof(CD)))

{

if(cost>n)

{

delay(50) ;

found = 1 ;

gotoxy(2,row) ;

cout<<cno;

gotoxy(14,row) ;

cout<<cname;

gotoxy(27,row) ;

cout<<sname;

gotoxy(42,row) ;

cout<<type;

gotoxy(55,row) ;

cout<<yr;

gotoxy(66,row) ;

cout<<nocopy;

gotoxy(74,row) ;

cout<<cost;

if ( row == 23 )

{

row=5 ;

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

pageno++ ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

clrscr() ;

cout<<"LIST OF CDS" ;

gotoxy(1,5) ;

cout<<"\nCD No. CD Name Singer's Name Type Release Sale Cost\n\n";

}

else

row++ ;

}

}

if ( !found )

{

gotoxy(1,5) ;

cout<<"Records not found" ;

}

gotoxy(66,1) ;

cout<<"Page no. : " <<pageno ;

gotoxy(1,25) ;

cout<<"Press any key to continue..." ;

getch() ;

file.close () ;

}

void backup()

{

char ch;

cout<<"\n Are You Sure That You Want To Create A Backup? (y/n)\n";

cin>>ch;

if(ch=='y')

{

CD e;

fstream f1,f2;

f1.open("CD.dat",ios::in|ios::binary);

f2.open("back.dat",ios::out|ios::binary);

while(f1.read((char\*)&e,sizeof(e)))

{

f2.write((char\*)&e,sizeof(e));

}

f1.close();

f2.close();

}

cout<<"\n\nBackup Created!!!!!!";

getch();

}

void restore()

{

char ch;

cout<<"\n Are You Sure You Want to Restore The Last Backup File? (y/n)\n";

cin>>ch;

if(ch=='y')

{

remove("CD.dat");

rename("back.dat","CD.dat");

cout<<"\n\nBackup File Restored....";

}

else

cout<<"\nFile not Restored!!!!\n\n";

}

void menu()

{

char ch1;

CD b;

int ch;

while(1)

{

clrscr();

cout<<"\n\n\n\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

cout<<"\n\t\t\t Search Menu\n";

cout<<"\n\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

cout<<"\t\t\t1. Search by CD Number\n\t\t\t2. Search by CD Name\n";

cout<<"\t\t\t3. Search by Singer Name\n\t\t\t4. Search by Release\n";

cout<<"\t\t\t5. Search by Type\n\t\t\t6. Search by Cost\n";

cout<<"\t\t\t7. Back to Previous Menu\n\t\t\t8. Exit\n";

cout<<"\n\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

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

gotoxy(35,22);

cin>>ch;

switch(ch)

{

case 1:

b.search();

break;

case 2:

b.searchcname();

break;

case 3:

b.shsname();

break;

case 4:

b.shrelease();

break;

case 5:

b.shtype();

break;

case 6:

b.shcost();

break;

case 7:

mainmenu();

break;

case 8:

exit(0);

default:

cout<<"\n\n\nInvalid Choice\n\n\n";

}

}

}

void mainmenu()

{

int ch;

char ch1;

CD b;

while(1)

{

clrscr();

cout<<"\n\n\n\n\n\n\n\n\n\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

cout<<"\t\t\*\*\*\*\*\* M . A . I . N - M . E . N . U \*\*\*\*\*\*\*\n";

cout<<"\n\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

cout<<"\n\t\t\t1. Add a Record\n\t\t\t2. Display All The Records\n";

cout<<"\t\t\t3. Modify a Record\n\t\t\t4. Search a Record\n";

cout<<"\t\t\t5. Delete a Record\n\t\t\t6. Create a Backup\n" ;

cout<<"\t\t\t7. Restore the Last Backup File\n\t\t\t8. Exit\n";

cout<<"\n\t\t \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

cout<<"\n\n\n\t\t Enter Your Choice: ";

gotoxy(38,29);

cin>>ch;

switch(ch)

{

case 1:

write();

break;

case 2:

b.read();

break;

case 3:

update();

break;

case 4:

menu();

break;

case 5:

del();

break;

case 6:

backup();

break;

case 7:

restore();

break;

case 8:

{

clrscr();

gotoxy(26,10);

delay(200);

cout<<"T";

delay(100);

cout<<"h";

delay(100);

cout<<"a";

delay(100);

cout<<"n";

delay(100);

cout<<"k";

delay(100);

cout<<" Y";

delay(50);

cout<<"o";

delay(100);

cout<<"u";

delay(100);

cout<<" F";

delay(100);

cout<<"o";

delay(100);

cout<<"r";

delay(100);

cout<<" V";

delay(100);

cout<<"i";

delay(100);

cout<<"s";

delay(100);

cout<<"i";

delay(100);

cout<<"t";

delay(100);

cout<<"i";

delay(100);

cout<<"n";

delay(100);

cout<<"g";

delay(50);

cout<<" M";

delay(100);

cout<<"y";

delay(100);

cout<<" P";

delay(100);

cout<<"r";

delay(100);

cout<<"o";

delay(100);

cout<<"j";

delay(100);

cout<<"e";

delay(100);

cout<<"c";

delay(100);

cout<<"t";

getch();

gotoxy(32,12);

cout<<" HAVE A NICE DAY " ;

getch();

exit(0);

}

default:

cout<<"\n\n\nInvalid Choice\n\n\n";

}

}

}

void main()

{

clrscr();

textcolor(GREEN);

char ch1;

int ch;

char pass[15];

gotoxy(30,15);

cout<<"Enter your Password : ";

gotoxy(55,15);

pass[0]=getch();

cout<<"\*";

pass[1]=getch();

cout<<"\*";

pass[2]=getch();

cout<<"\*";

pass[3]=getch();

cout<<"\*";

pass[4]=getch();

cout<<"\*";

pass[5]=getch();

cout<<"\*";

pass[6]=getch();

cout<<"\*";

pass[7]='\0';

getch();

CD b;

if(strcmpi(pass,"ojassvi")==0)

{

clrscr();

cout<<"\n\n\n\n\n\t\t\t\tWELCOME TO MY PROJECT \n\n\n\n\t\t\t\tCD Management System\n\n\n\n\t\t\t\tMUSIC WORLD";

cout<<"\n\n\n\t\t\tMade by:";

delay(100);

cout<<"O";

delay(100);

cout<<"J";

delay(100);

cout<<"A";

delay(100);

cout<<"S";

delay(100);

cout<<"S";

delay(100);

cout<<"V";

delay(100);

cout<<"I";

delay(100);

cout<<" ";

delay(100);

cout<<"P";

delay(100);

cout<<"R";

delay(100);

cout<<"A";

delay(100);

cout<<"D";

delay(100);

cout<<"H";

delay(100);

cout<<"A";

delay(100);

cout<<"N";

gotoxy(25,25);

cout<<" Press any Key to Continue\n";

getch();

mainmenu();

}

else

cout<<"\n\n\n\nInvalid Password!!!!!!Try Again.....\n\n\n";

getch();

}

OUTPUT

SCREEN -1:

Enter your Password : \*\*\*\*\*\*\*

SCREEN -2:

WELCOME TO MY PROJECT CD Management System

MUSIC WORLD

Made by: OJASSVI PRADHAN Press any Key to Continue

SCREEN -3:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\* M . A . I . N - M . E . N . U \*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. Add a Record

2. Display All The Records

3. Modify a Record

4. Search a Record

5. Delete a Record

6. Create a Backup

7. Restore the Last Backup File

8. Exit \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter Your Choice:

SCREEN -4:

Enter The CD Number:- 2

Enter The CD Title:- Paradise Enter Singer's Name:- Taylor Enter The Type of The CD:- mp3 Enter No. of Copies:- 7 Enter Year of Release:- 2013 Enter the Cost of CD:- 299 RECORD ADDED SUCCESSFULLY!!!!!!!!!!

SCREEN -5:

LIST OF CDS Page no. : 1

CD No. CD Name Singer's Name Type Release Sale Cost

1 Mirrors Timber Audio 2009 3 500

2 Pradise Taylor mp3 2013 7 299

Press any key to continue...

SCREEN -6:

Enter CD No. to be Modified

2

Enter New Values

Enter The CD Title:- House

Enter Singer's Name:- Lady Gaga

Enter The Type of The CD:- Audio

Enter No. of Copies:- 9

Enter Year of Release:- 2010

Enter the Cost of The CD:- 99

Record Modified Successfully

LIST OF CDS Page no. : 1

CD No. CD Name Singer's Name Type Release Sale Cost

1 Mirrors Timber Audio 2009 3 500

2 House Lady Gaga Audio 2010 9 99

Press any key to continue...

SCREEN -7:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* . Search Menu \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. Search by CD Number

2. Search by CD Name

3. Search by Singer Name

4. Search by Release

5. Search by Type

6. Search by Cost

7. Back to Previous Menu

8. Exit \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter your choice:

SCREEN -8:

LIST OF CDS Page no. : 1

CD No. CD Name Singer's Name Type Release Sale

Cost

1 Mirrors Timber Audio 2009 3 500

Press any key to continue...

SCREEN -9:

Enter the CD name to be searched:

OLD

LIST OF CDS Page no. : 1

CD No. CD Name Singer's Name Type Release Sale

Cost

2 House Lady Gaga Audio 2010 9 99

Press any key to continue...

SCREEN -10:

Enter the Singer’s name whose CDs to be searched:

Timber

LIST OF CDS Page no. : 1 CD No. CD Name Singer's Name Type Release Sale

Cost

1 Mirrors Timber Audio 2009 3 500

Press any key to continue...

SCREEN -11:

Enter the Year of Release of the CD to be searched:

2009

LIST OF CDS Page no. : 1 CD No. CD Name Singer's Name Type Release Sale

Cost

1 Mirrors Timber Audio 2009 3 500

Press any key to continue...

SCREEN -12:

Enter the type of the CD to be searched:

Audio

LIST OF CDS Page no. : 1 CD No. CD Name Singer's Name Type Release Sale

Cost

1 Mirrors Timber Audio 2009 3 500

2 House Lady Gaga Audio 2010 9 99

Press any key to continue...

SCREEN -13:

Enter the minimum cost to see the CDs:

30

LIST OF CDS Page no. : 1 CD No. CD Name Singer's Name Type Release Sale

Cost

1 Mirrors Timber Audio 2009 3 500

2 House Lady Gaga Audio 2010 9 99

Press any key to continue...

SCREEN -14:

Enter CD Number to be Deleted

1

Record Deleted!!!!!!!

LIST OF CDS Page no. : 1 CD No. CD Name Singer's Name Type Release Sale

Cost

2 House Lady Gaga Audio 2010 9 99

Press any key to continue…

SCREEN -15:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\* M . A . I . N - M . E . N . U \*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. Add a Record

2. Display All The Records

3. Modify a Record

4. Search a Record

5. Delete a Record

6. Create a Backup

7. Restore the Last Backup File

8. Exit \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter Your Choice:6

Are you sure you want to create a backup??(y/n)

y

Backup Created!!!!!!!!!!!!!!!

SCREEN -16:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\* M . A . I . N - M . E . N . U \*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. Add a Record

2. Display All The Records

3. Modify a Record

4. Search a Record

5. Delete a Record

6. Create a Backup

7. Restore the Last Backup File

8. Exit \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Enter Your Choice:7

Are You Sure You Want to Restore The Last Backup File? (y/n)

y

Backup File Restored....

SCREEN -17:

Thank You For Visiting My Project HAVE A NICE DAY