**DATA STRUCTURES LAB PROJECT**

15B17CI371

Title - Video Library

Submitted by-

Anuranjan Dubey (9917103247)

Pranjal Das (9917103244)

Abhishek Kumar Saxena (9917103246)

Krishna Ballav Narayan Singh (9917103260)

Supervised by-

Sudhanshu Kulshrestha



**Department of CSE/IT**

**Jaypee Institute Of Information Technology, Noida**

**September 2018Language used:-**

C++

**Data structures and concepts used:-**

1. Array

2. File handling

3. Linked list

4. Binary search tree

5. OOPS(classes and objects)

**About the project:-**

It's a video library management system. It has the following options

1) Album issue - Issues a customer album of his/her choice.

2) Album deposit- Submits the album isuued to the respective customer.

3)Administrator menu -

i) Create customer record

ii) Display all customer record

iii) Display specific customer record

iv) Modify customer record

v) Delete customer record

vi) Create new album

vii) Display all albums

viii) Display specific album

ix) Modify album

x) Delete album

4) Operations on Sorting of album- Sorts the index no.'s of albums using BST

5) Exit - To exit

**Functions used :-**

1) main() - Show the name of project and asks for password . Validates the

password and show the functions mentioned above for the user

to acess.

2) write\_Video\_Index()- This function is used to add albums.

3) display\_spb(char n[])- This function show details of album whose index is

entered by the user.

4) modify\_Video\_Index() - This function allows the user to modify the details

of the desired album.

5)delete\_Video\_Index() - This allows the user to delete particular album using

it's index no.

6) display\_allb() - This function is used to display all the albums.

7) Video\_Index\_issue() - This function is used to issue a particular album to a

particular album.

8) Video\_Index\_deposit() - This function is used to deposit a album by

customer and it also calculates fine upon the no.

days exceeded .

9) write\_customer() - This function is used to add customer's record.

10) display\_sps(char n[])- This function show details of customer whose customer no. is entered by the user.

11) modify\_customer() -This function allows the user to modify the details of the respective customer.

12) delete\_customer() -This allows the user to delete particular customer's details using it's index no.

13)display\_alls() - - This function is used to display all the customer record.

14) BSToprtr() - This function is used to sort the index no.'s of albums.

15) admin\_menu() - Displays all the menu options as mentioned under

Administrator menu.

16) intro() - Gives the intro of the project .

**Contributions :-**

Pranjal and Krishna - All the functions related to album management :-

1) write\_Video\_Index()

2) display\_spb(char n[])

3) modify\_Video\_Index()

4)delete\_Video\_Index()

5) display\_allb()

6) Video\_Index\_issue()

7) Video\_Index\_deposit()

8)admin menu()

Abhishek and Anuranjan - All the functions related to customer record

management and BST sorting function.

1) write\_customer()

2) display\_sps(char n[])

3) modify\_customer()

4) delete\_customer()

5)display\_alls()

6) BSToprtr()

7) intro()

8) main()

Source code:

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// HEADER FILE USED IN PROJECT

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include<iostream>

#include<fstream>

#include<conio.h>

#include<stdlib.h>

#include<stdio.h>

#include<process.h>

#include<string.h>

#include<iomanip>

using namespace std;

void admin\_menu();

struct node

{

int key;

struct node \*left,\*right;

};

struct node \*newnode(int key)

{

struct node \*temp=new node;

temp->key=key;

temp->left=temp->right=NULL;

return temp;

};

void inorder(struct node \*temp)

{

if (!temp)

return;

inorder(temp->left);

cout<<temp->key<<" ";

inorder(temp->right);

}

struct node\* insart(struct node \*temp,int key)

{

if (temp==NULL)

return newnode(key);

if (key<temp->key)

temp->left=insart(temp->left,key);

else if (key>temp->key)

temp->right=insart(temp->right,key);

return temp;

}

void BSToprtr()

{

int nr,key;

char charac;

struct node \*root=newnode(0);system("cls");

cout<<"\n\n\n\t\tNote indexing of album index starts from '0' "<<endl;

cout<<"\t\tEnter the number of records:";

cin>>nr;

for (int z=0;z<nr;z++)

{

cout<<"\n\t\tEnter the unsorted album indexes.:";

cin>>key;

insart(root, key);

cout << "\n\t\tInorder traversal of BST after insertion of given unsorted album index is.: ";

inorder(root);

} cout<<"\n\t\tPress any key to go to admin menu"<<endl;

getch();

admin\_menu();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// CLASS USED IN PROJECT

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class Video\_Index

{

char bno[6];

char bname[50];

char aname[20];

public:

void create\_Video\_Index()

{

cout<<"\nNEW VIDEO ENTRY MENU...\n";fflush(stdin);

cout<<"\nEnter The Album Index no.:";fflush(stdin);

cin>>bno;fflush(stdin);

cout<<"\n\nEnter The Name of The Album.: ";fflush(stdin);

gets(bname);fflush(stdin);

cout<<"\n\nEnter The Author's Name..: ";fflush(stdin);

gets(aname);fflush(stdin);

cout<<"\n\n\nAlbum Added Successfully..";fflush(stdin);

}

void show\_Video\_Index()

{

cout<<"\nAlbum Index no. : "<<bno;

cout<<"\nAlbum Name. : ";

puts(bname);

cout<<"Author Name. : ";

puts(aname);

}

void modify\_Video\_Index()

{

cout<<"\nAlbum Index no. : "<<bno;fflush(stdin);

cout<<"\nModify Album Name. : ";fflush(stdin);

gets(bname);fflush(stdin);

cout<<"\nModify Author's Name of Album. : ";

gets(aname);fflush(stdin);

}

char\* retbno()

{

return bno;

}

void report()

{cout<<bno<<setw(30)<<bname<<setw(30)<<aname<<endl;}

}; //class ends here

class customer

{

char admno[6];

char name[20];

char stbno[6];

int token;

public:

void create\_customer()

{

system("cls");

cout<<"\nNEW ALBUM ENTRY MENU...\n";fflush(stdin);

cout<<"\nEnter The Customer no. :";fflush(stdin);

cin>>admno;fflush(stdin);

cout<<"\n\nEnter The Name of The customer. :";fflush(stdin);

gets(name);fflush(stdin);

token=0;

stbno[0]='/0';

cout<<"\n\nCustomer Record Created Successfully...";

}

void show\_customer()

{

cout<<"\nCustomer no. : "<<admno;fflush(stdin);

cout<<"\nCustomer Name. : ";fflush(stdin);

puts(name);fflush(stdin);

cout<<"\nNo of Album issued. : "<<token;

if(token==1)

cout<<"\nAlbum No.:"<<stbno;

}

void modify\_customer()

{

cout<<"\nCustomer no. : "<<admno;

cout<<"\nModify customer Name. : ";fflush(stdin);

gets(name);fflush(stdin);

}

char\* retadmno()

{

return admno;

}

char\* retstbno()

{

return stbno;

}

int rettoken()

{

return token;

}

void addtoken()

{token=1;}

void resettoken()

{token=0;}

void getstbno(char t[])

{

strcpy(stbno,t);

}

void report()

{cout<<"\t"<<admno<<setw(20)<<name<<setw(10)<<token<<endl;}

}; //class ends here

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// global declaration for stream object, object

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

fstream fp,fp1;

Video\_Index bk;

customer st;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// function to write in file

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void write\_Video\_Index()

{

char ch;

fp.open("Video\_Index.dat",ios::out|ios::app);

do

{

system("cls");

bk.create\_Video\_Index();

fp.write((char\*)&bk,sizeof(Video\_Index));

cout<<"\n\nDo you want to add more record..(y/n?)";fflush(stdin);

cin>>ch;fflush(stdin);

}while(ch=='y'||ch=='Y');

fp.close();

}

void write\_customer()

{

char ch;

fp.open("customer.dat",ios::out|ios::app);

do

{

st.create\_customer();

fp.write((char\*)&st,sizeof(customer));

cout<<"\n\ndo you want to add more record..(y/n?)";fflush(stdin);

cin>>ch;fflush(stdin);

}while(ch=='y'||ch=='Y');

fp.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// function to read specific record from file

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void display\_spb(char n[])

{

cout<<"\nALBUM DETAILS...;\n";

int flag=0;

fp.open("Video\_Index.dat",ios::in);

while(fp.read((char\*)&bk,sizeof(Video\_Index)))

{

if(strcmpi(bk.retbno(),n)==0)

{

bk.show\_Video\_Index();

flag=1;

}

}

fp.close();

if(flag==0)

cout<<"\n\nAlbum does not exist";

getch();

}

void display\_sps(char n[])

{

cout<<"\nCUSTOMER DETAILS\n";

int flag=0;

fp.open("customer.dat",ios::in);

while(fp.read((char\*)&st,sizeof(customer)))

{

if((strcmpi(st.retadmno(),n)==0))

{

st.show\_customer();

flag=1;

}

}

fp.close();

if(flag==0)

cout<<"\n\nCustomer does not exist";

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// function to modify record of file

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void modify\_Video\_Index()

{

char n[6];

int found=0;

system("cls");

cout<<"\n\n\tMODIFY ALBUM REOCORD.... ";fflush(stdin);

cout<<"\n\n\tEnter The Album Index no.:";fflush(stdin);

cin>>n;fflush(stdin);

fp.open("Video\_Index.dat",ios::in|ios::out);

while(fp.read((char\*)&bk,sizeof(Video\_Index)) && found==0)

{

if(strcmpi(bk.retbno(),n)==0)

{

bk.show\_Video\_Index();

cout<<"\nEnter The New Details of Album:"<<endl;

bk.modify\_Video\_Index();

int pos=-1\*sizeof(bk);

fp.seekp(pos,ios::cur);

fp.write((char\*)&bk,sizeof(Video\_Index));

cout<<"\n\n\t Record Updated";

found=1;

}

}

fp.close();

if(found==0)

cout<<"\n\n Record Not Found ";

getch();

}

void modify\_customer()

{

char n[6];

int found=0;

system("cls");

cout<<"\n\n\tMODIFY CUSTOMER RECORD... ";fflush(stdin);

cout<<"\n\n\tEnter The Index no. of The customer:";fflush(stdin);

cin>>n;fflush(stdin);

fp.open("customer.dat",ios::in|ios::out);

while(fp.read((char\*)&st,sizeof(customer)) && found==0)

{

if(strcmpi(st.retadmno(),n)==0)

{

st.show\_customer();

cout<<"\nEnter The New Details of customer:"<<endl;

st.modify\_customer();

int pos=-1\*sizeof(st);

fp.seekp(pos,ios::cur);

fp.write((char\*)&st,sizeof(customer));

cout<<"\n\n\t Record Updated";

found=1;

}

}

fp.close();

if(found==0)

cout<<"\n\n Record Not Found ";

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// function to delete record of file

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void delete\_customer()

{

char n[6];

int flag=0;

system("cls");

cout<<"\n\n\n\tDELETE CUSTOMER RECORD...";fflush(stdin);

cout<<"\n\nEnter The Index no. of the customer You Want To Delete : ";fflush(stdin);

cin>>n;fflush(stdin);

fp.open("customer.dat",ios::in|ios::out);

fstream fp2;

fp2.open("Temp.dat",ios::out);

fp.seekg(0,ios::beg);

while(fp.read((char\*)&st,sizeof(customer)))

{

if(strcmpi(st.retadmno(),n)!=0)

fp2.write((char\*)&st,sizeof(customer));

else

flag=1;

}

fp2.close();

fp.close();

remove("customer.dat");

rename("Temp.dat","customer.dat");

if(flag==1)

cout<<"\n\n\tRecord Deleted ..";//fflush(stdin);

else

cout<<"\n\nRecord not found";

getch();

}

void delete\_Video\_Index()

{

char n[6];

system("cls");

cout<<"\n\n\n\tDELETE ALBUM MENU...";fflush(stdin);

cout<<"\n\nEnter The Album Index no. You Want To Delete : ";fflush(stdin);

cin>>n;fflush(stdin);

fp.open("Video\_Index.dat",ios::in|ios::out);

fstream fp2;

fp2.open("Temp.dat",ios::out);

fp.seekg(0,ios::beg);

while(fp.read((char\*)&bk,sizeof(Video\_Index)))

{

if(strcmpi(bk.retbno(),n)!=0)

{

fp2.write((char\*)&bk,sizeof(Video\_Index));

}

}

fp2.close();

fp.close();

remove("Video\_Index.dat");

rename("Temp.dat","Video\_Index.dat");

cout<<"\n\n\tRecord Deleted ..";

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// function to display all customers list

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void display\_alls()

{

system("cls");

fp.open("customer.dat",ios::in);

if(!fp)

{

cout<<"ERROR!!! FILE COULD NOT BE OPEN ";

getch();

return;

}

cout<<"\n\n\t\tcustomer LIST\n\n";

cout<<"==================================================================\n";

cout<<"\tAdmission No."<<setw(10)<<"Name"<<setw(20)<<"Video\_Index Issued\n";

cout<<"==================================================================\n";

while(fp.read((char\*)&st,sizeof(customer)))

{

st.report();

}

fp.close();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// function to display Video\_Indexs list

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void display\_allb()

{

system("cls");

fp.open("Video\_Index.dat",ios::in);

if(!fp)

{

cout<<"ERROR!!! FILE COULD NOT BE OPEN ";

getch();

return;

}

cout<<"\n\n\t\tVideo\_Index LIST\n\n";

cout<<"=========================================================================\n";

cout<<"Video\_Index Number"<<setw(20)<<"Video\_Index Name"<<setw(25)<<"Author\n";

cout<<"=========================================================================\n";

while(fp.read((char\*)&bk,sizeof(Video\_Index)))

{

bk.report();

}

fp.close();

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// function to issue Video\_Index

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void Video\_Index\_issue()

{

char sn[6],bn[6];

int found=0,flag=0;

system("cls");

cout<<"\n\nALBUM ISSUE MENU...";fflush(stdin);

cout<<"\n\n\tEnter The customer's index no.";fflush(stdin);

cin>>sn;fflush(stdin);

fp.open("customer.dat",ios::in|ios::out);

fp1.open("Video\_Index.dat",ios::in|ios::out);

while(fp.read((char\*)&st,sizeof(customer)) && found==0)

{

if(strcmpi(st.retadmno(),sn)==0)

{

found=1;

if(st.rettoken()==0)

{

cout<<"\n\n\tEnter the Album Index no. ";

cin>>bn;

while(fp1.read((char\*)&bk,sizeof(Video\_Index))&& flag==0)

{

if(strcmpi(bk.retbno(),bn)==0)

{

bk.show\_Video\_Index();

flag=1;

st.addtoken();

st.getstbno(bk.retbno());

int pos=-1\*sizeof(st);

fp.seekp(pos,ios::cur);

fp.write((char\*)&st,sizeof(customer));

cout<<"\n\n\t Album issued successfully\n\nPlease Note: Write the current date in backside of your Album and submit within 15 days fine Rs. 1 for each day after 15 days period";

}

}

if(flag==0)

cout<<"Album does not exist";

}

else

cout<<"You have not returned the last Album ";

}

}

if(found==0)

cout<<"customer record not exist...";

getch();

fp.close();

fp1.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// function to deposit Video\_Index

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void Video\_Index\_deposit()

{

char sn[6],bn[6];

int found=0,flag=0,day,fine;

system("cls");

cout<<"\n\nALBUM DEPOSIT MENU...";

cout<<"\n\n\tEnter The customerís index no.";fflush(stdin);

cin>>sn;fflush(stdin);

fp.open("customer.dat",ios::in|ios::out);

fp1.open("Video\_Index.dat",ios::in|ios::out);

while(fp.read((char\*)&st,sizeof(customer)) && found==0)

{

if(strcmpi(st.retadmno(),sn)==0)

{

found=1;

if(st.rettoken()==1)

{

while(fp1.read((char\*)&bk,sizeof(Video\_Index))&& flag==0)

{

if(strcmpi(bk.retbno(),st.retstbno())==0)

{

bk.show\_Video\_Index();

flag=1;

cout<<"\n\nAlbum deposited in no. of days.:";

cin>>day;

if(day>15)

{

fine=(day-15)\*1;

cout<<"\n\nFine has to deposited Rs.:"<<fine;

}

st.resettoken();

int pos=-1\*sizeof(st);

fp.seekp(pos,ios::cur);

fp.write((char\*)&st,sizeof(customer));

cout<<"\n\n\t Album deposited successfully";

}

}

if(flag==0)

cout<<"Album does not exist";

}

else

cout<<"No Album is issued..please check!!";

}

}

if(found==0)

cout<<"Customer record does not exist...";

getch();

fp.close();

fp1.close();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// INTRODUCTION FUNCTION

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void intro()

{

system("cls");

cout<<"\n\n\n\t\t\t\t----------LIBRARY MANAGEMENT SYSTEM----------";

cout<<"\n\n\t\t\t\t\tMADE BY .: Pranjal Das(9917103244)";

cout<<"\n\n\t\t\t\t\t : Abhishek Kumar Saxena(9917103246)";

cout<<"\n\n\t\t\t\t\t : Anuranjan Dubey(9917103247)";

cout<<"\n\n\t\t\t\t\t : Krishna Ballav Narayan Singh(9917103260)";

cout<<"\n\n\n\n\t\t\t\t\t\t JIIT-128";

getch();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// ADMINISTRATOR MENU FUNCTION

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void admin\_menu()

{

system("cls");

int ch2;

cout<<"\n\n\n\tADMINISTRATOR MENU";

cout<<"\n\n\t1.CREATE CUSTOMER RECORD";

cout<<"\n\n\t2.DISPLAY ALL CUSTOMER RECORD";

cout<<"\n\n\t3.DISPLAY SPECIFIC CUSTOMER RECORD ";

cout<<"\n\n\t4.MODIFY CUSTOMER RECORD";

cout<<"\n\n\t5.DELETE CUSTOMER RECORD";

cout<<"\n\n\t6.CREATE NEW ALBUM ";

cout<<"\n\n\t7.DISPLAY ALL ALBUMS ";

cout<<"\n\n\t8.DISPLAY SPECIFIC ALBUMS ";

cout<<"\n\n\t9.MODIFY ALBUM ";

cout<<"\n\n\t10.DELETE ALBUM ";

cout<<"\n\n\t11.BACK TO MAIN MENU";

cout<<"\n\n\tPlease Enter Your Choice (1-11) ";

cin>>ch2;

switch(ch2)

{

case 1: system("cls");

write\_customer();break;

case 2: display\_alls();break;

case 3:

char num[6];

system("cls");

cout<<"\n\n\tPlease Enter The Customer No.: ";fflush(stdin);

cin>>num;fflush(stdin);

display\_sps(num);

break;

case 4: modify\_customer();break;

case 5: delete\_customer();break;

case 6: system("cls");

write\_Video\_Index();break;

case 7: display\_allb();break;

case 8: {

char num[6];

system("cls");

cout<<"\n\n\tPlease Enter The Video\_Index No. ";fflush(stdin);

cin>>num;fflush(stdin);

display\_spb(num);

break;

}

case 9: modify\_Video\_Index();break;

case 10: delete\_Video\_Index();break;

case 11: return;

default:cout<<"\a";

}

admin\_menu();

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// THE MAIN FUNCTION OF PROGRAM

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

int main()

{

system("color B");

char ch;

intro();

char pass[10],password[10]="jiitian";

char arr[10];

printf ("\n\n\n\t\t\t\t Enter master password to login.:");

for (int i=0;i<7;i++)

{

arr[i]=getch();

putch('\*');

}

if (arr[0]=='j' && arr[1]=='i' && arr[2]=='i' && arr[3]=='t' && arr[4]=='i' && arr[5]=='a' && arr[6]=='n')

{

printf ("\n");

printf ("\n\n\n\n\t\t\t\t\t\t Access granted!");

cout<<"\n\n\t\t\t\t\t Press any key to continue";

getch();

system("cls");

//menu();

do

{

system("cls");

cout<<"\n\n\n\tMAIN MENU";

cout<<"\n\n\t01. ALBUM ISSUE";

cout<<"\n\n\t02. ALBUM DEPOSIT";

cout<<"\n\n\t03. ADMINISTRATOR MENU";

cout<<"\n\n\t04. OPERATIONS ON SORTING OF ALBUM INDEX(BST)";

cout<<"\n\n\t05. EXIT";

cout<<"\n\n\tPlease Select Your Option (1-5) \n\n\n\n";

ch=getche();fflush(stdin);

switch(ch)

{

case '1':system("cls");

Video\_Index\_issue();

break;

case '2':Video\_Index\_deposit();

break;

case '3':admin\_menu();

break;

case '4':BSToprtr();//function responsible for operations in BST;

break;

case '5':return(0);

default :cout<<"\a";

}

}while(ch!='5');

}

else

{

printf ("\n\n\n\n\t\t\t\tYou just entered wrong password Access Denied!!\n\n\n");

}

}