**CONTENTS**

1. Certificate
2. Acknowledgement
3. Header files and their purpose
4. Some particular coding
5. Outputs

**HEADER FILES**

1. **FSTREAM.H** – for file handling, cin and cout
2. **PROCESS.H** – for exit() function
3. **CONIO.H** – for clrscr() and getch() functions
4. **STDIO.H** – for standard I/O operations
5. **STRING.H** – for string handling
6. **IOMANIP**.**H**- for setw() function
7. **DOS**.**H**- for delay() function

**SOME CODING**

* **FOR CREATE STUDENT:-**

void write\_student()

{

char ch;

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

do

{

st.create\_student();

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

cout<<"\n\n\t\tDO YOU WANT TO ADD MORE RECORD..(y/n?): ";

ch=getche();

delay(500);

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

fp.close();

}

* **FOR CREATE BOOK:-**

void write\_book()

{

char ch;

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

do

{

clrscr();

bk.create\_book();

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

cout<<"\n\n\t\tDO YOU WANT TO ADD MORE RECORD..(y/n?): ";

ch=getche();

delay(500);

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

fp.close();

}

* **FOR MODIFY STUDENT:-**

void modify\_student()

{

char n[6];

int found=0;

clrscr();

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

cout<<"\n\t\t ÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»";

cout<<"\n\t\t º MODIFY STUDENT RECORD º";

cout<<"\n\t\t º º";

cout<<"\n\t\t º STUDENT ADMISSION NO : º";

cout<<"\n\t\t ÈÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ¼";

gotoxy(43,14);

cin>>n;

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

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

{

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

{

st.show\_student();

cout<<"\n\n\t\t ENTER THE NEW DETAILS OF STUDENT"<<endl;

st.modify\_student();

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

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

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

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

found=1;

}

}

fp.close();

if(found==0)

cout<<"\n\n\t\t \*\*\*\*\*RECORD NOT FOUND\*\*\*\*\* ";

getch();

}

* **FOR MODIFY BOOK:-**

void modify\_book()

{

char n[6];

int found=0;

clrscr();

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

cout<<"\n\t\t ÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»";

cout<<"\n\t\t º MODIFY BOOK RECORD º";

cout<<"\n\t\t º º";

cout<<"\n\t\t º BOOK NO : º";

cout<<"\n\t\t ÈÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ¼";

gotoxy(30,14);

cin>>n;

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

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

{

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

{

bk.show\_book();

cout<<"\n\n\t\t ENTER THE NEW DETAILS OF BOOK"<<endl;

bk.modify\_book();

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

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

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

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

found=1;

}

}

fp.close();

if(found==0)

cout<<"\n\n\t\t \*\*\*\*\*RECORD NOT FOUND\*\*\*\*\* ";

getch();

}

* **FOR DELETE STUDENT:-**

void delete\_student()

{

char n[6];

int flag=0;

clrscr();

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

cout<<"\n\t\t ÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»";

cout<<"\n\t\t º DELETE STUDENT º";

cout<<"\n\t\t º º";

cout<<"\n\t\t º STUDENT ADMISSION NO : º";

cout<<"\n\t\t ÈÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ¼";

gotoxy(43,14);

cin>>n;

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

fstream fp2;

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

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

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

{

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

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

else

flag=1;

}

fp2.close();

fp.close();

remove("student.dat");

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

if(flag==1)

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

else

cout<<"\n\n\t\t \*\*\*\*\*RECORD NOT FOUND\*\*\*\*\*";

getch();

}

* **FOR DELETE BOOK:-**

void delete\_book()

{

char n[6];

int flag=0;

clrscr();

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

cout<<"\n\t\t ÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»";

cout<<"\n\t\t º DELETE BOOK º";

cout<<"\n\t\t º º";

cout<<"\n\t\t º BOOK NO : º";

cout<<"\n\t\t ÈÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ¼";

gotoxy(30,14);

cin>>n;

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

fstream fp2;

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

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

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

{

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

{

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

}

else

flag=1;

}

fp2.close();

fp.close();

remove("book.dat");

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

if(flag==1)

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

else

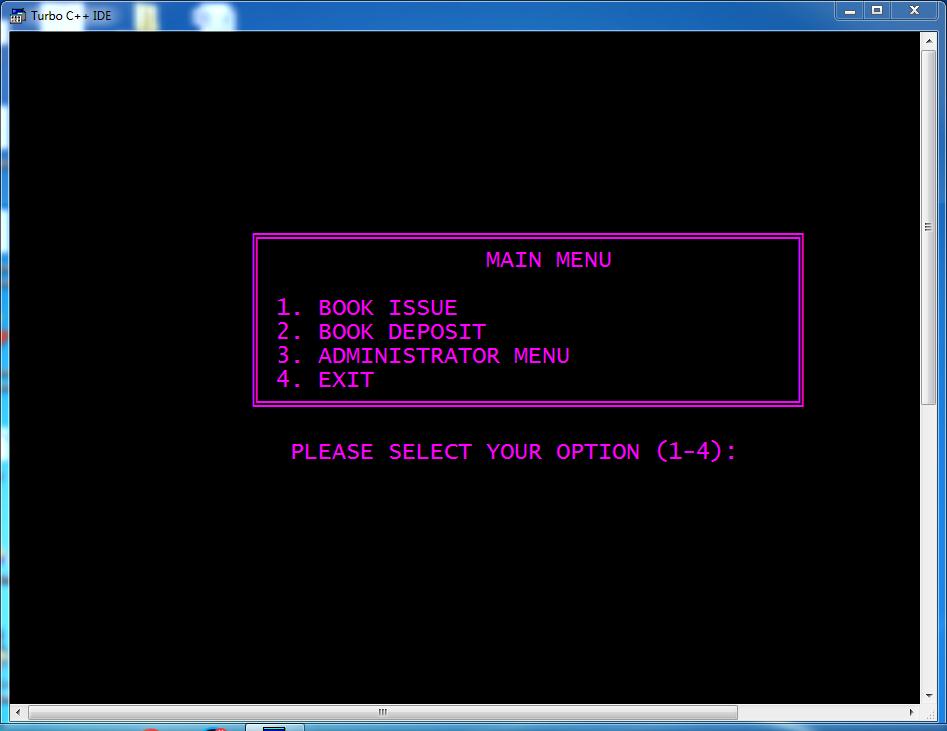
cout<<"\n\n\t\t \*\*\*\*\*RECORD NOT FOUND\*\*\*\*\*";

getch();

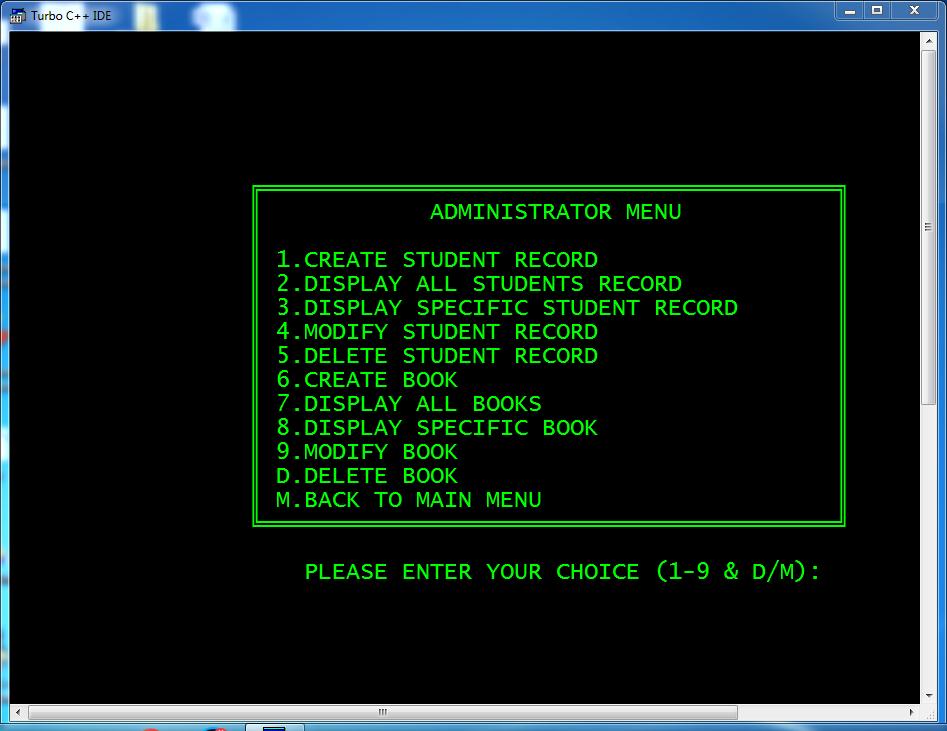
}

**OUTPUTS**

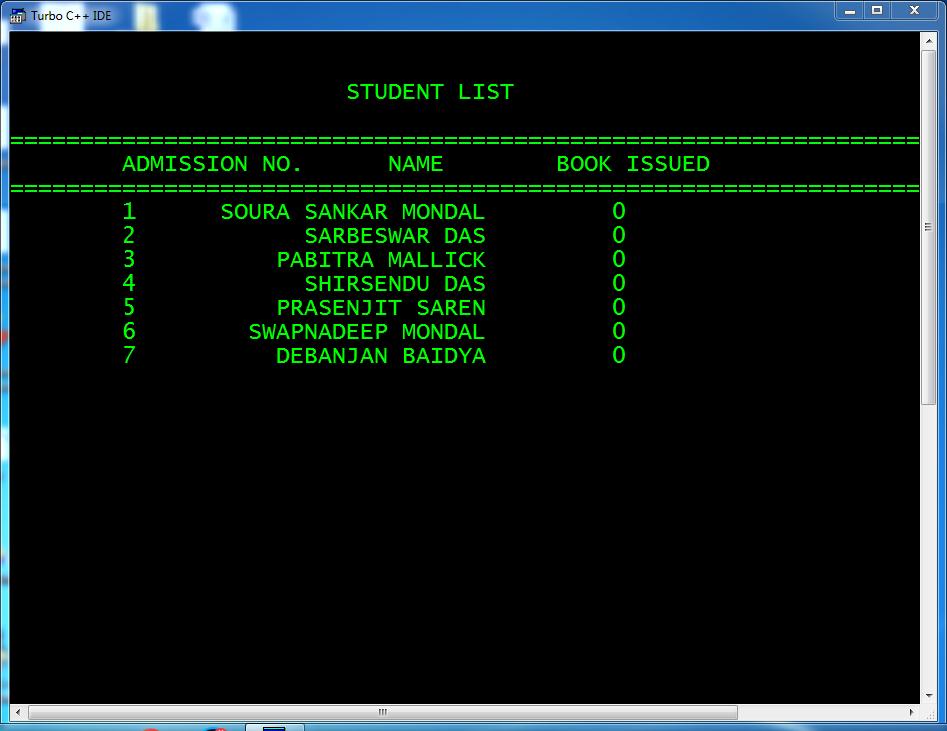
* **FOR MAIN MENU:-**

****

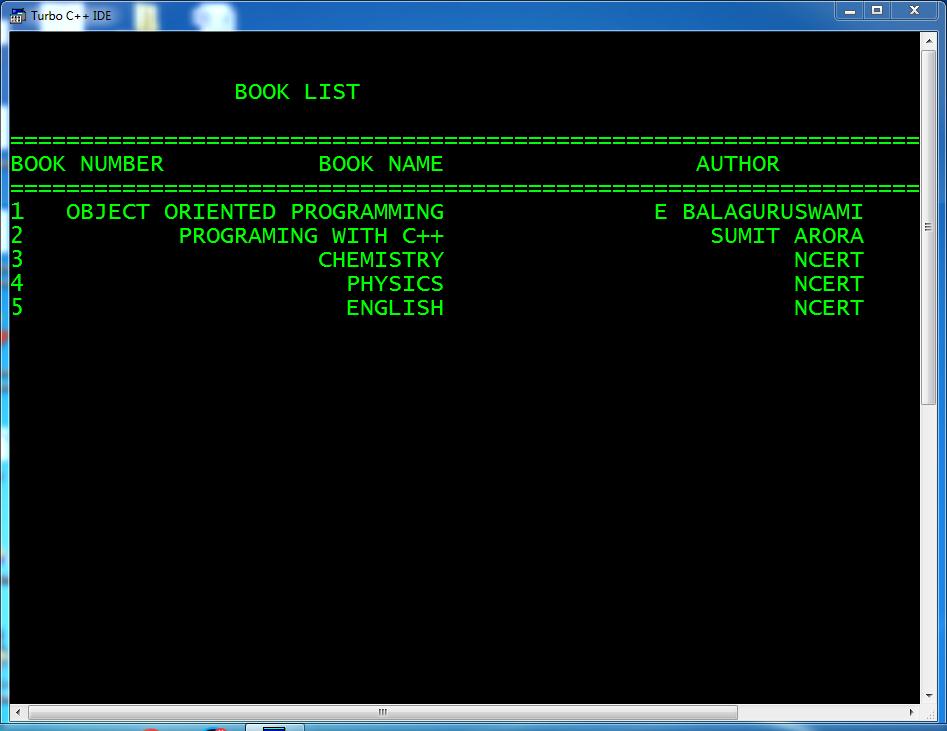
* **FOR ADMINISTRATOR MENU:-**

****

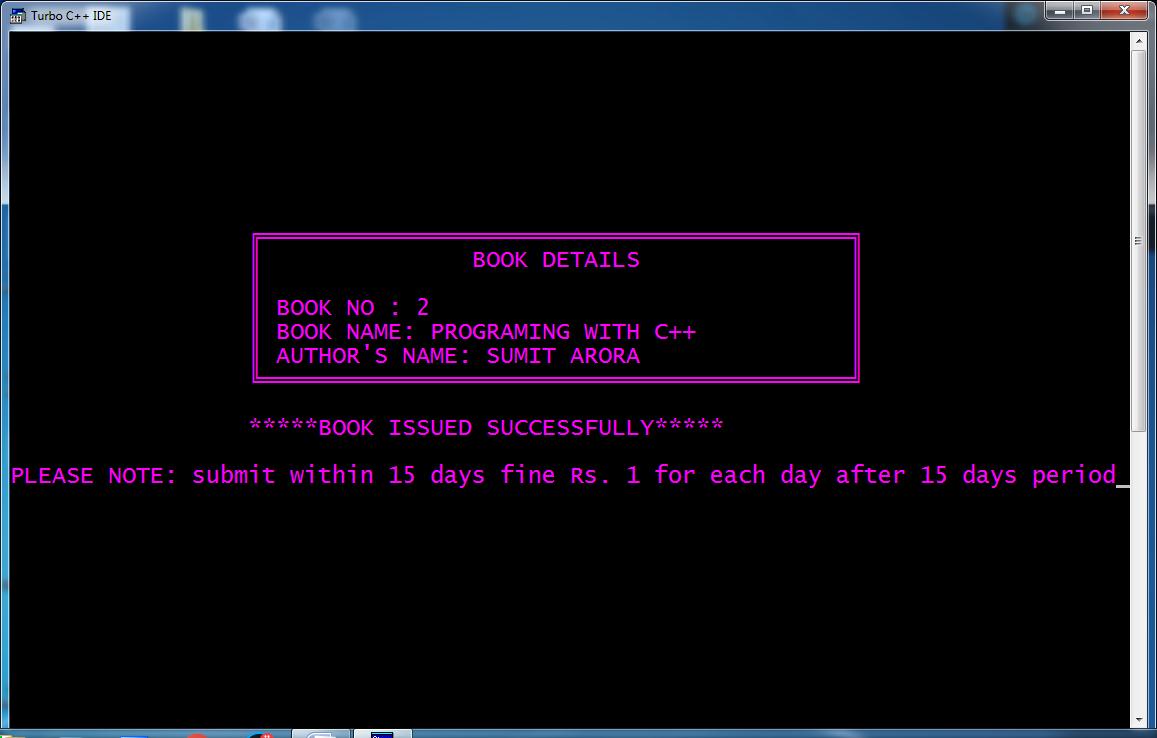
* **FOR DISPLAY ALL STUDENTS:-**

****

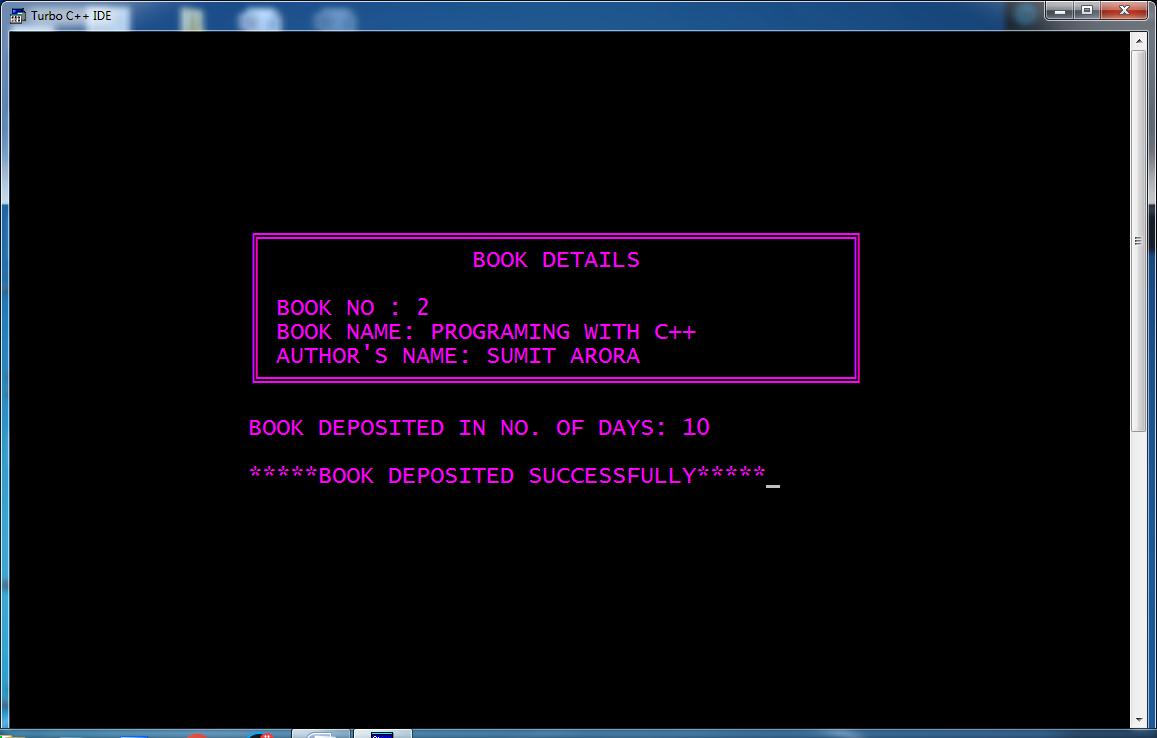
* **FOR DISPLAY ALL BOOK:-**

****

* **FOR BOOK ISSUE:-**



* **FOR DEPOSITE BOOK:-**



***THANKS***