INTRODUCTION

A highly populated school that is where a lot of students are studying means a lot of visitors come to school like the parents of the students, school officials etc. So it's very important to keep a track on the movements on each and every visitor but it's not actually feasible for the schools. But at least we can keep a track of the people who entered in the school for the sake of security of the school.

This program is designed to keep the Visitors record in the School. This program is very useful in real life situation for providing instant information of the Visitors who have visited the school. It also stores their general information such as their **NAME, AGE, and PHONE NUMBER** etc.

THE SCHOOL VISITOR RECORD MANAGEMENT aims to make a simpler staff's interaction with the various modules of the program for recording the information of the visitors and ease the process of acquiring information and providing services. The system can be accessed by the **ADMINISTRATOR** and **USER** but the highest priority given to **ADMINISTRATOR** that
Are allocated a login passkey. It will also allow cutting the insecurity of the school due to visitors.

In this system we will make extensive use of files system in C++.We will have a separate priorities for both the **ADMINISTRATOR** and **USERS**. If you are a user you can access the general module of the program which includes adding record, searching record etc. But if you access as Administrator then you can access the more secure module that is the log book in which the record of the users are kept that is who used this system , at what time etc. which increases the security of the school and the system itself . An Overview of the Program is as follows:

This Program consists of FIVE Functions as follows:

- **CREATING NEW RECORD FOR THE VISITOR**
- **VIEWING ALL THE RECORDS**
- **SEARCHING RECORD**
- **UPDATING RECORD**
- **DELETING RECORD**
- **EXITING THE PROGRAM**

In this software developed separate functions will be there for each of the above points so that there is ample scope for adding more features in the future.

HEADER FILES USED

1. **IOSTREAM.H** – for file handling, cin and cout

2. **CONIO.H** – for getch ()

3. **PROCESS.H** – for exit ()

4. **FSTREAM.H** – for file stream handling

5. **STRING.H –** for string handling

6. **STDIO.H** – for standard I/O functions

8. **TIME.H –** for time and date functions

FILES GENERATED

DATA FILES

- LOGBOOK.DAT
- VISITOR.DAT

PROGRAM FILES

- SCHOOL VISITOR RECORD.CPP

EXECUTION FILE

- SCHOOL VISITOR RECORD.EXE

CODING

```
#include<iostream>
using namespace std;
#include<conio.h>
#include<process.h>
#include<fstream>
#include<string.h>
#include<stdio.h>
#include<time.h>

int permit=0;
class user //For Users
{
        char uname[10],rel[10],paskey[10],timee_1[10],datte_1[10];
        public:
                void get()//getting the details of the user
                {
                _strtime(timee_1);
                _strdate(datte_1);
                cout<<"\nENTER USER'S NAME:";
                cin>>uname;
                cout<<"\nENTER YOUR RELATION WITH THE SCHOOL:";
                cin>>rel;
                }
                void print()//showing the details of the user
                {
                        cout<<"\nDATE:"<<datte_1;
                        cout<<"\nTIME:"<<timee_1;
                        cout<<"\nUSER'S NAME:"<<uname;
                        cout<<"\nRELATION WITH SCHOOL:"<<rel;
                        cout<<"\n-------------------------------------\n\n\n";
                }
                int check_pskey(char pas[])//matching the passkeys
                {
                        if(strcmp(pas,"admin")==0)
                        return 1;
                        else
                        return 0;
                }
}u;//Object Name
class visitor//For Visitors
{
int visno,age;
char datte_in[10],timee_in[10],name[25],addr[25];
char phone[25],plate[10],purpose[25],email[25],status[10];
char datte_out[10],timee_out[10];
public:
void input()//Input Member Function
```

```
{
                        _strtime(timee_in);
                        _strdate(datte_in);
                       cout<<"\nNAME:";
                       cin>>name;
                       cout<<"\nAGE:";
                       cin>>age;
                       cout<<"\nADDRESS:";
                       cin>>addr;
                       cout<<"\nCONTACT NUMBER:";
                       cin>>phone;
                       cout<<"\nVEHICLE PLATE NUMBER:";
                       cin>>plate;
                       cout<<"\nPURPOSE OF VISIT:";
                       cin>>purpose;
                       cout<<"\nEMAIL-ID:";
                       cin>>email;
                       strcpy(status,"STILL_INSIDE");
}
void update()//updating the record of the visitor
{
         strcpy(status,"WENT_OUT");
        _strtime(timee_out);
        _strdate(datte_out);
        cout<<"\nYOUR DATABASE HAS BEEN SUCCESSFULLY UPDATED ....\n";
        cout<<"\nTHE VISITOR HAS BEEN MARKED AS GONE OUT ....\n";
        cout<<"\n@@@@@@@@@@ THANKS FOR VISITING @@@@@@@@@@@\n";
}
void show()    //Output Member Function
{
                       cout<<"\nDATE IN                          :"<<datte_in;
                       cout<<"\nTIME IN                          :"<<timee_in;
                       cout<<"\nNAME                             :"<<name;
                       cout<<"\nAGE                              :"<<age;
                       cout<<"\nADDRESS                          :"<<addr;
                       cout<<"\nCONTACT NUMBER                        :"<<phone;
                       cout<<"\nVEHICLE PLATE NUMBER                  :"<<plate;
                       cout<<"\nPURPOSE OF VISIT                 :"<<purpose;
                       cout<<"\nEMAIL-ID                  :"<<email;
                       cout<<"\nSTATUS                   :"<<status;
                       if(permit=1)
                       {
                       cout<<"\nDATE OUT                   :"<<datte_out;
                       cout<<"\nTIME OUT                   :"<<timee_out;
                       }

        cout<<"\n\t\t****************************************"<<endl<<endl<<endl;
}
void modify()//To Modify The Record
{
```

```
char cnfrm='Y',newname[25],newphone[25],newaddr[25],newplate[25],newpurp[25],newemail[25];
        int newage;
        cout<<endl;
        cout<<"YOUR OLD DATABASE CONTAINS:"<<endl<<endl;
        show();
        cout<<endl<<endl;
        cout<<"\nPRESS ANY KEY TO CONTINUE...";
        getch();
        system("cls");
        cout<<endl<<endl;
        cout<<"\nENTER THE NEW DATABASE:"<<endl;
        cout<<"\nENTER NEW NAME:";
        cin>>newname;
        cout<<"\nENTER NEW AGE:";
        cin>>newage;
        cout<<"\nENTER NEW ADDRESS:";
        cin>>newaddr;
        cout<<"\nENTER NEW CONTACT NUMBER:";
        cin>>newphone;
        cout<<"\nENTER NEW VEHICLE PLATE NUMBER:";
        cin>>newplate;
        cout<<"\nENTER NEW PURPOSE OF VISIT:";
        cin>>newpurp;
        cout<<"\nENTER NEW EMAIL-ID:";
        cin>>newemail;
        cout<<"\nPRESS Y TO CONFIRM THE CHANGES AND ANY OTHER KEY TO
DISCARD THE CHANGES...";
        cin>>cnfrm;
        if(cnfrm=='Y'||cnfrm=='y')//After Getting Confirmation
        {
                age=newage;
                strcpy(name,newname);
                strcpy(addr,newaddr);
                strcpy(phone,newphone);
                strcpy(plate,newplate);
                strcpy(purpose,newpurp);
                strcpy(email,newemail);
                cout<<"\nYOUR DATABASE HAS BEEN SUCCESSFULLY UPDATED\n\n";
        }
        else
cout<<"\nMODIFICATION PROCESS CANCELLED !!\nPRESS ANY KEY TO CONTINUE...";
cout<<"\nPRESS ANY KEY TO VIEW THE CURRENT DATABASE:";
        getch();
        system("cls");
        cout<<"\nNOW THE CURRENT DATABASE CONTAINS THE FOLLOWING DATA:";
        cout<<endl;
        show();
}
int ag()//To return the age for searching
{
        return age;
}
```

```
int checkdate(char b[])//Compare the Date Of Visit
{
	if(strcmpi(b,datte_in)==0)
	return 1;
	else
	return 0;
}
int checkmonth(char c[])//Compare the Month Of Visit
{
	int p=0,q=0,r=0,s=0,st;
	for(int i=0;i<=1;i++)
	{
		if(datte_in[i]==c[i])
		p++;
	}
	if(p==2)
	r=1;
	for(int j=6;j<=7;j++)
	{
		if(datte_in[j]==c[j])
		q++;
	}
	if(q==2)
	s=1;
	st=r+s;
	return st;
}
int checkyear(char d[])//Compare the Year Of Visit
{
	int t=0,stts_1=0;
	for(int j=7;j<=8;j++)
	{
		if(datte_in[j]==d[j]);
		t++;
	}
	if(t==2)
	stts_1=1;
	return stts_1;
}
int nam(char a[])//Function To Compare Names
{
	if(strcmpi(a,name)==0)
	return 1;
	else
	return 0;
}
}me;//Object Name
 void head()//Shows Program Heading
 {
		cout<<"\n\t\t      SCHOOL VISITOR RECORD MANAGEMENT\n";
		cout<<"\n\t\t    ------------------------------------\n\n\n";
 }
```

```
 void skipper()//Asks To Go To Main Menu
 {
        cout<<"\nPRESS ENTER TO GO TO MAIN MENU...";
        getch();
        system("cls");
 }
int main()//Beggining of main()
{
cout<<"\n\n\n\t\t\t  KENDRIYA VIDYALAYA D.G.Q.A\n";
cout<<"\n\t\t\t ***************************\n";
cout<<"\n\n\t\t    COMPUTER SCIENCE INVESTIGATORY PROJECT\n";
cout<<"\n\t\t    ------------------------------------\n";
head();
cout<<"\nPRESS ENTER TO CONTINUE....";
getch();
system("cls");
head();
cout<<"\n\n\nPREPARED BY:\n\n";
cout<<"\n1.ASHISH KUMAR PUNIA \n2.DEV ASHISH SHARMA \n3.RUDRAKSH CHANDRA
MUKUT \n\n";
cout<<"\nPRESS ENTER TO CONTINUE...";
getch();
int ch1,ch3,ch4,ch5;char ch2='y';
system("cls");
head();
ofstream qw;
qw.open("logbook.dat",ios::app);
while(qw)
{
        u.get();
        qw.write((char*)&u,sizeof(u));
        break;
}
qw.close();
cout<<"\nLOG ADDED TO FILE .... \n\n\n";
cout<<"\nPRESS ENTER TO CONTINUE...";
getch();
first:
system("cls");
head();
cout<<"\n USE THIS SOFTWARE AS ....\n\n";
cout<<"\n1.ADMINISTRATOR \n2.USER";
cout<<"\n\n\nENTER YOUR CHOICE:";
cin>>ch3;
if(ch3==1)
{
        int ch3_1;char pskey[10];
        askpassagain:
        cout<<"\nENTER THE PASSKEY:";
        cin>>pskey;
        if(u.check_pskey(pskey)==1)
        {
```

```
        system("cls");
        head();
        cout<<"\nLOGGED IN AS ADMINISTRATOR...\n\n\n";
        cout<<"\n1.VIEW LOG \n2.GO TO PREVIOUS MENU \n3.EXIT THE PROGRAM";
        cout<<"\n\n\nENTER YOUR CHOICE:";
        cin>>ch3_1;
        if(ch3_1==1)
        {
                ifstream qwe;
                qwe.open("logbook.dat",ios::in);
                while(qwe)
                {
                        qwe.read((char*)&u,sizeof(u));
                        u.print();
                }
                qwe.close();
        }
        else if(ch3_1==2)
        goto first;
        else
        exit(4);
        }
        else
        {
                char psopt_1;
cout<<"\nINVLAID PASSKEY!! \nPRESS y TO TRY AGAIN AND n TO GO TO PREVIOUS
MENU...";
                cin>>psopt_1;
                if(psopt_1=='y')
                goto askpassagain;
                else
                goto first;
        }
}
else
{
main:
system("cls");
head();
cout<<"\nLOGGED IN AS USER...\n\n\n";
cout<<"\n1.CREATE RECORD";
cout<<"\n2.VIEW ALL RECORDS";
cout<<"\n3.SEARCH RECORD";
cout<<"\n4.MODIFY RECORD";
cout<<"\n5.DELETE RECORD";
cout<<"\n6.EXIT\n";
cout<<"\nCHOOSE YOUR OPTION:";
cin>>ch1;
if(ch1==1)//Writing New or Fresh Records
{
        system("cls");
        head();
```

```
        cout<<"\nWRITING NEW RECORD .....\n\n\n";
        ofstream qw;
        qw.open("visitor.dat",ios::app);
        while(toupper(ch2)=='Y')
        {
                me.input();
                qw.write((char*)&me,sizeof(me));
                cout<<"\nPRESS y TO ENTER MORE RECORDS...";
                cin>>ch2;
        }
        qw.close();
        cout<<"\n\nPRESS ENTER TO GO TO MAIN MENU...\n";
        getch();
        goto main;
}
else if(ch1==2)//Viewing All The Record Present In The File
{
        system("cls");
        head();
        cout<<"\nVIEWING ALL THE RECORDS....\n\n\n";
        ifstream qwe;
        qwe.open("visitor.dat",ios::in);
        while(qwe)
        {
                qwe.read((char*)&me,sizeof(me));
                me.show();
                cout<<"\nPRESS ANY KEY FOR NEXT RECORD ...";
                cout<<endl;
                cout<<endl;
                getch();
        }
        cout<<"\n\nNO RECORDS ARE LEFT !!";
        skipper();
        qwe.close();
        goto main;
}
else if(ch1==3)//Searching Records
{
        int ch3;
        main2:
        system("cls");
        head();
        cout<<"\n\t\t\t  SEARCHING THE VISITOR \n\t\t\t _____\n\n\n";
        cout<<"\n1.SEARCH RECORD NY NAME";
        cout<<"\n2.SEARCH RECORD BY AGE";
        cout<<"\n3.SEARCH RECORD BY SPECIFIC DATE OF VISIT";
        cout<<"\n4.SEARCH RECORD BY SPECIFIC MONTH OF VISIT";
        cout<<"\n5.SEARCH RECORD BY SPECIFIC YEAR OF VISIT";
        cout<<"\n6.GO TO MAIN MENU";
        cout<<"\nENTER YOUR CHOICE..."<<endl;
        cin>>ch3;
        if(ch3==1)//Searching Record By The Name Of The Visitor
```

```
	{
		char nm[25];
		system("cls");
		try_again_2:
		ifstream qwe;
		qwe.open("visitor.dat",ios::in);
		head();
		cout<<"\nSEARCHING RECORD BY THE NAME OF THE VISITOR ...\n\n\n";
		cout<<"\nENTER THE NAME OF THE VISITOR TO BE SEARCHED FOR:";
		cin>>nm;
		char status_2='n';
		qwe.seekg(ios::beg);
		while(!qwe.eof())
		{
			qwe.read((char*)&me,sizeof(me));
			if(me.nam(nm)==1)
			{
				status_2='y';
				me.show();
			}
		}
		skipper();
		goto main;
		if(status_2=='n')
		{
		cout<<"\nTHE RECORD YOU REQUESTED IS NOT FOUND!!";
		cout<<"\nPRESS y TO TRY AGAIN AND ANY OTHER KEY TO EXIT...";
		cin>>status_2;
		if(status_2=='y')
		goto try_again_2;
			}
		qwe.close();
	}
	else if(ch3==2)//Searching Record By Age Of The Visitor
	{
		int agge;
		try_again_1:
		system("cls");
		ifstream qwe;
		qwe.open("visitor.dat",ios::in);
		head();
cout<<"\nSEARCHING RECORD BY THE AGE OF THE VISITOR...\n\n\n";
cout<<"\nENTER THE AGE OF THE VISITOR WHOSE RECORD IS TO BE SEARCHED:";
		cin>>agge;
		char status_3='n';
		qwe.seekg(ios::beg);
		while(!qwe.eof())
		{
			qwe.read((char*)&me,sizeof(me));
			if(me.ag()==agge)
			{
				status_3='y';
```

```
                        me.show();
                    }
                }
                if(status_3=='n')
                {
                cout<<"\nTHE RECORD YOU REQUESTED IS NOT FOUND!!";
                cout<<"\nPRESS y TO TRY AGAIN AND ANY OTHER KEY TO EXIT...";
                cin>>status_3;
                if(status_3=='y')
                goto try_again_1;
                }
                qwe.close();
        }
        else if(ch3==3)//Searching Record By Date Of Visit
        {
                char date_1[10],opt_2;
                try_again_3:
                system("cls");
                ifstream qwe;
                qwe.open("visitor.dat",ios::in);
                qwe.seekg(ios::beg);
                head();
                cout<<"\nSEARCHING RECORD BY THE DATE OF VISIT...\n\n\n";
                cout<<"\nENTER THE DATE OF VISITING WHICH IS TO BE SEARCHED:";
                cout<<"\nDATE:";cin>>date_1;
                        while(qwe)
                        {
                        qwe.read((char*)&me,sizeof(me));
                                if(me.checkdate(date_1)==1)
                                {
                                 me.show();
                                }
                        else
                        {
                        cout<<"\nRECORD NOT FOUND !! PRESS y TO TRY AGAIN AND
ANY OTHER KEY TO EXIT...";
                        cin>>opt_2;
                        if(opt_2=='y')
                        goto try_again_3;
                                }
                    }
                qwe.close();
        }
        else if (ch3==4)//Searching Record By Month Of Visit
        {
                char month[10],opt_3;
                try_again_4:
                system("cls");
                ifstream qwe;
                qwe.open("visitor.dat",ios::in);
                qwe.seekg(ios::beg);
                head();
```

```
		cout<<"\nSEARCHING RECORD BY THE MONTH OF VISIT...\n\n\n";
		cout<<"\nENTER THE MONTH OF VISITING WHICH IS TO BE SEARCHED:";
		cout<<"\nMONTH:";
		cin>>month;
				  while(qwe)
				{
				qwe.read((char*)&me,sizeof(me));
					if(me.checkmonth(month)==1)
					{
					 me.show();
					}
				else
				{
cout<<"\nRECORD NOT FOUND !! PRESS y TO TRY AGAIN AND ANY OTHER KEY TO
EXIT...";
				cin>>opt_3;
				if(opt_3=='y')
				goto try_again_4;
					}
						}
		qwe.close();
	}
	else if(ch3==5)//Searching Record By Year
	{
		char year[10],opt_4;
		try_again_5:
		system("cls");
		ifstream qwe;
		qwe.open("visitor.dat",ios::in);
		qwe.seekg(ios::beg);
		head();
		cout<<"\nSEARCHING THE RECORD BY THE YEAR OF VISIT...\n\n\n";
		cout<<"\nENTER THE YEAR OF VISITING WHICH IS TO BE SEARCHED:";
		cout<<"\nYEAR:";
		cin>>year;
				while(qwe)
				{
				qwe.read((char*)&me,sizeof(me));
					if(me.checkyear(year)==1)
					{
					 me.show();
					}
				else
				{
cout<<"\nRECORD NOT FOUND !!\nPRESS y TO TRY AGAIN AND ANY OTHER KEY TO
EXIT...";
				cin>>opt_4;
				if(opt_4=='y')
				goto try_again_5;
					}
						}
		qwe.close();
```

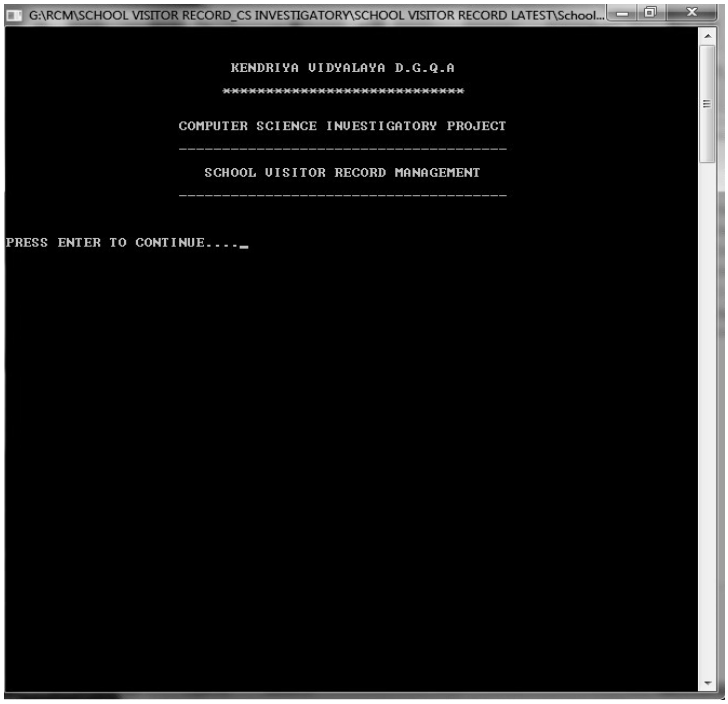
```
        }
        else
        goto main;
        }
        else if (ch1==4)//For Modifying A Record
        {
        long pos_1;
        char status_4='n',modnm[25],run_1;
        int ch;
        fstream q;
        q.open("visitor.dat",ios::in|ios::out);
        round_1:
        system("cls");
        head();
        cout<<"\n1.MODIFY THE WHOLE RECORD \n2.SET THE TIME OUT\n\n";
        cout<<"\nENTER YOUR CHOICE:";
        cin>>ch;
        if(ch==1)//modifying the whole record
        {
        system("cls");
        head();
cout<<"\nMODIFYING RECORD BY NAME OF THE VISITOR AS THE SEARCH
PARAMETER...\n\n\n";
cout<<"\nENTER THE NAME OF THE VISITOR WHOSE RECORD IS TO BE MODIFIED:";
        cin>>modnm;
        fstream q;
        q.open("visitor.dat",ios::in|ios::out);
        while(!q.eof())
        {
                pos_1=q.tellg();
                q.read((char*)&me,sizeof(me));
                if(me.nam(modnm)==1)//Searching Record By Name Of The Visitor
                {
                        me.modify();
                        q.seekg(pos_1);
                        q.write((char*)&me,sizeof(me));
                        status_4='y';
                        break;
                }
                }
        if(status_4=='n')
        cout<<"\nRECORD NOT FOUND!!";
        cout<<"\nPRESS y TO TRY AGAIN AND ANY OTHER KEY TO EXIT...\n";
        cin>>run_1;
        if(run_1=='y')
        goto round_1;
        else
        goto skip_1;
        q.seekg(0);
        system("cls");
        head();
        cout<<"\nTHE RECORD HAS BEEN MODIFIED ... \n\nNOW THE FILE CONTAINS:\n";
```

```
        while(!q.eof())
        {
                q.read((char*)&me,sizeof(me));
                me.show();
        }
        cout<<endl<<endl;
        cout<<"\nPRESS ANY KEY TO FINISH...";
        getch();
        skip_1:
        q.close();
        }
        else//setting the time out
        {
                long pos_2;
                char modnm_1[25],status_5='n',run_2='n';
                system("cls");
                head();
cout<<"\nSETTING THE TIME OUT OF THE VISITOR ....\n\n\n";
cout<<"\nENTER THE NAME OF THE VISITOR WHOSE RECORD IS TO BE MODIFIED:";
                cin>>modnm_1;
                fstream q;
                q.open("visitor.dat",ios::in|ios::out);
                        while(!q.eof())
                                {
                                        pos_2=q.tellg();
                                        q.read((char*)&me,sizeof(me));
                                if(me.nam(modnm_1)==1)//Searching Record By Name Of The Visitor
                                        {
                                                me.update();
                                                q.seekg(pos_2);
                                                q.write((char*)&me,sizeof(me));
                                                status_5='y';
                                                permit=1;
                                        }
                                }
        if(status_5=='y')
        {
        cout<<"\nPRESS ANY KEY TO GO TO MAIN MENU ....\n";
        getch();
        goto main;
        }
        if(status_5=='n')
        {
        cout<<"\nRECORD NOT FOUND!!";
        cout<<"\nPRESS y TO TRY AGAIN AND ANY OTHER KEY TO EXIT...\n";
        cin>>run_2;
        }
        if(run_2=='y')
        goto round_1;
        q.close();
        }
        }
```

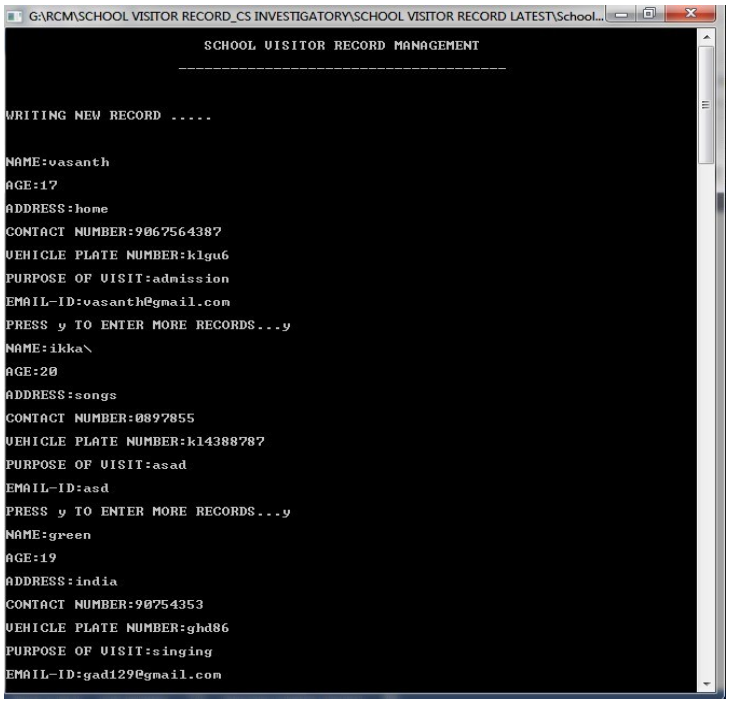
```
        else if(ch1==5)//For Deleting A Record
        {
        ifstream qw;
        qw.open("visitors.dat",ios::in);
        ofstream qwe;
        qwe.open("temp.dat",ios::out);
        char delnm[25];
        char found='f',confirm='n';
        system("cls");
        head();
cout<<"\nDELETING RECORD BY NAME OF THE VISITOR AS SEARCH
PARAMETER...\n\n\n";
cout<<"ENTER THE NAME OF THE VISITOR WHOSE RECORD IS TO BE DELETED:";
        cin>>delnm;
        while(!qw.eof())
        {
        qw.read((char*)&me,sizeof(me));
        if(me.nam(delnm)==1)
        {
         me.show();
         found='t';
         cout<<"\nARE YOU SURE YOU WANT TO DELETE THIS RECORD??";
         cin>>confirm;
         if(confirm=='n')
         qwe.write((char*)&me,sizeof(me));
        }
        else
        qwe.write((char*)&me,sizeof(me));
    }
        if(found=='f')
        cout<<"\nTHE RECORD YOU REQUESTED IS NOT FOUND!!!";
        qw.close();
        qwe.close();
        system("del visitor.dat");
        system("ren temp.dat visitor.dat");
        qw.open("visitor.dat",ios::in);
        cout<<"\nNOW THE FILE CONTAINS...";
        while(!qw.eof())
        {
                qw.read((char*)&me,sizeof(me));
                cout<<"\nPRESS ANY KEY FOR NEXT RECORD...";
                getch();
                me.show();
        }
}
else
exit(5);//For exiting the Program
getch();
return 0;
}
        }//End of main()
```

OUTPUT SCREENS

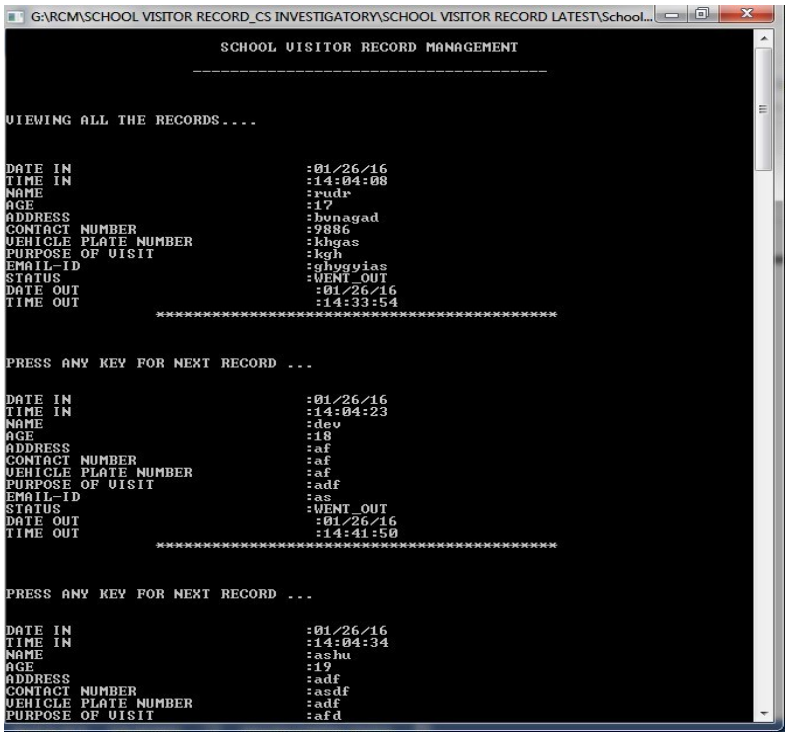
1. WELCOME SCREEN



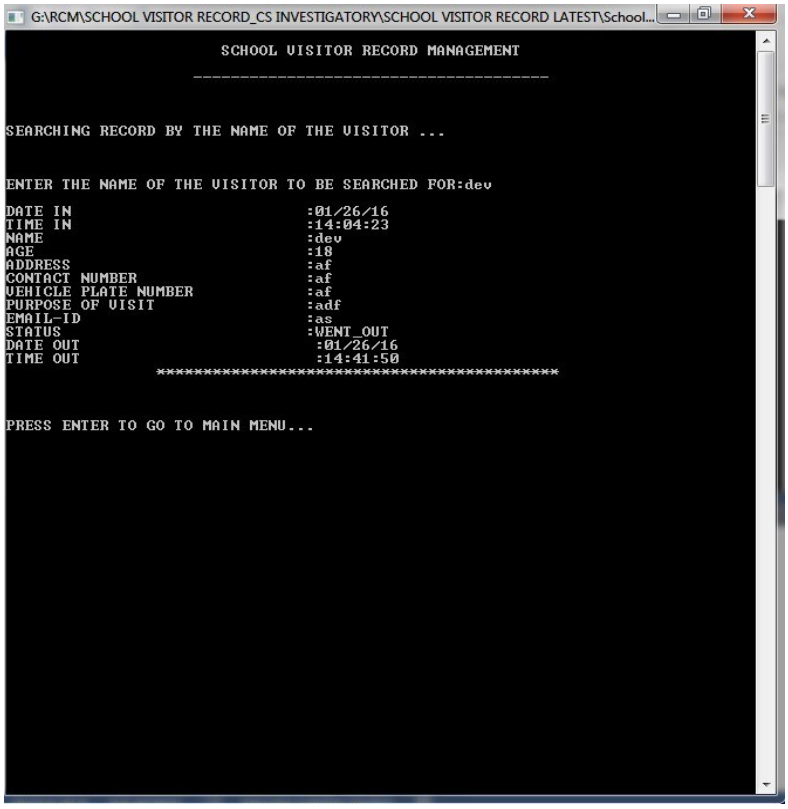
2. ADDING NEW RECORD



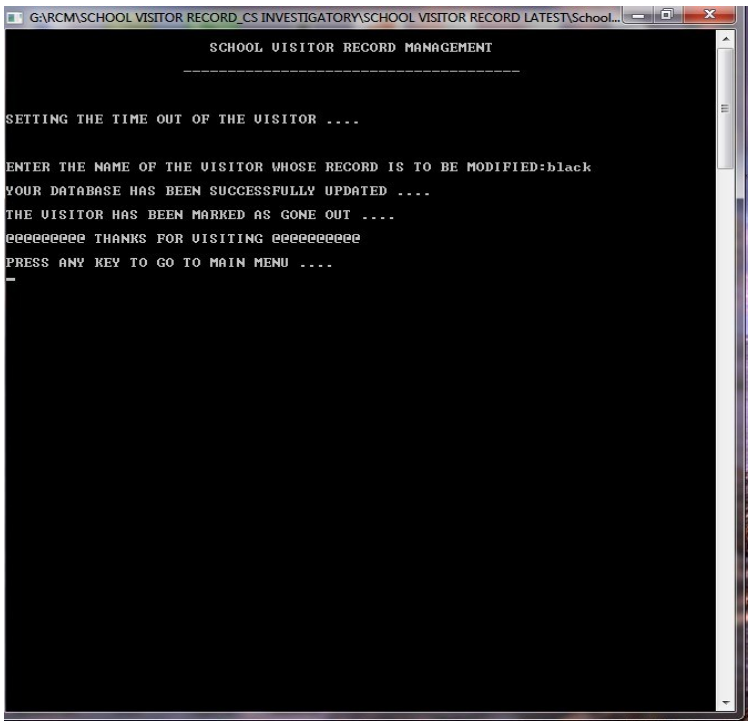
2. VIEWING ALL RECORDS



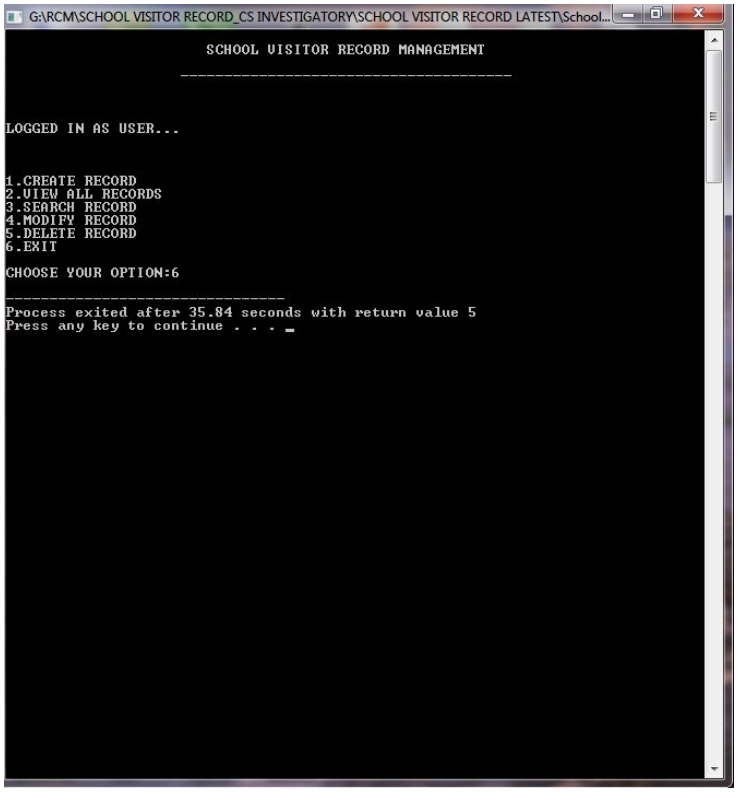
4. SEARCH RECORD



5. UPADATING DATA



6. EXIT



FUTURE SCOPE

As this software is desktop version, it can be run on standalone system. The data/record can be viewed only on single desktop on which the software is installed. In future the code may be converted into web application which can be viewed/monitored by various concern authorities, through LAN on their system, however the data entry/bio-metric inputs will be done only one system at entrance/main gate.

CONCLUSION

At the end of this coursework, I was able to:

- Explain **OBJECT ORIENTED PROGRAMING CONCEPTS** and apply them to the modeling of real world systems.

- Explain the **OBJECT ORIENTED** paradigm and utilization of the offered facilities.

- Demonstrate the ability to develop and derive new class structures and organize them such that they will model the real world system within computers.

BIBLIOGRAPHY

- CLASS NOTES
- REFRENCE BOOKS – INTRODUCTION TO C++ BY **SUMITA ARORA**

THANK YOU

I would like to thank my computer science sir **MR. PREM KUMAR SINGH** , who provided us this oppurtinity to make a project on **SCHOOL VISITOR RECORD MANAGEMENT** which really helped us to understand the concepts of C++ Programming.