

COMPUTER SCIENCE

TOPIC: SCHOOL MANAGEMENT SYSTEM

RIFAATH MOHAMED AMEEN

XII - N

ACKNOWLEDGMENT

IN THE ACKNOWLEDGMENT OF THIS PROJECT SUCCESSFULLY, MANY PEOPLE HAVE BEST OWNED UPON ME THEIR HEART PLEDGED SUPPORT.

PRIMARILY, I WOULD LIKE TO THANK GOD FOR BEING ABLE TO COMPLETE THIS PROJECT WITH SUCCESS. THEN I WOULD LIKE TO THANK MY PRINCIPAL DR. NASREEN BANU AND MY COMPUTER SCIENCE TEACHER MRS. DHIVYA ANOOP, WHOSE VALUABLE GUIDANCE HAS BEEN THE ONES THAT HELPED ME PATCH THIS PROJECT. HER SUGGESTIONS AND INSTRUCTIONS HAVE SERVED AS THE MAJOR CONTRIBUTOR TO THE COMPLETION OF THIS PROJECT.

THEN I WOULD LIKE TO THANK MY PARENTS AND FRIENDS WHO HAVE HELPED ME WITH THEIR VALUABLE SUGGESTIONS AND GUIDANCE HAS BEEN VERY HELPFUL IN VARIOUS PHASES OF THE COMPLETION OF THE PROJECT.

FINALLY, I WOULD LIKE TO THANK MY CLASSMATES WHO HAVE HELPED ME A LOT.

INDEX

- 1. INTRODUCTION**
- 2. HARDWARE AND SOFTWARE REQUIREMENTS**
- 3. DESIGN**
- 4. SOURCE CODE**
- 5. OUTPUT**
- 6. CONCLUSION**
- 7. BIBLIOGRAPHY**

INTRODUCTION

School management system plays an essential role in the current educational system. School authorities all over the world are engaged in a lot of day-to-day administrative and academic activities to manage and provide a better academic experience to students effectively. However, maintaining and keeping track of school administrative activities is not an easy process in the fast-growing world. It requires hard work and often it is time-consuming.

To better perform the school administrative activities of educational institute, a well-maintained database of all administrative records is necessary.

This project is about maintaining a database with the records of all teachers, bus fleet and students, and includes several options, procedures for each module with complete file concept.

HARDWARE & SOFTWARE REQUIREMENTS

Software required:

- Turbo C++ 3.0 or above / Borland C++

Hardware required:

- Intel Pentium & CPU 3.06 GHz
- 3.08 GHz, 4gb ram
- 1,44 FDD
- 40gb HDD
- 52x CDR
- 32m VGA card
- Windows 7 or above

DESIGN

This program is designed to keep records of teachers, students, and bus fleet.

This program consists of three options with 3 internal options as follows:

1. To add/remove/search teacher record
2. To add/remove/search student record
3. To add/remove/search bus record

SOURCE CODE

```
#include <iostream.h>
#include <conio.h>
#include <dos.h>
#include <stdlib.h>
#include <string.h>
#include <constrea.h>
#include <stdio.h>
#include <process.h>
#include <fstream.h>
#include <cctype.h>
#include <sys/stat.h>
#include <dir.h>

char title[28] = "µSCHOOL MANAGEMENT SYSTEMÆ";

void line(int xi = 0, int yi = 0, int xf = 79, int yf = 0, int speed = 75,
int style = 0)
{
    int xdel,ydel;
    if(speed!=0)
    {
        xdel = 100/speed;
        ydel = 200/speed;
    }
    else
    {
```

```

        xdel=0;ydel=0;
    }

style%=4;

char sym[4][2] = {
    {'Í', 'º'},
    {'Ä', '³'},
    {'Ü', 'Ý'},
    {'±', '±'}
};

if(xi == xf){
    for(int i = yi; i<=yf; i++){
        gotoxy(xi, i);
        cout<<sym[style][1];
        delay(ydel);
    }
}

else{
    for(int i = xi; i<=xf; i++){
        gotoxy(i, yi);
        cout<<sym[style][0];
        delay(xdel);
    }
}

}

void box(int xpos = 0, int ypos = 0, int xsize = 77, int ysize = 24, int
speed = 50, int style = 0)

```

```

{

char sym[6][8] = {
{'É', 'Í', '»', 'º', '%', 'Í', 'È', 'º'},
{'Ú', 'Ä', 'ç', '³', 'Ù', 'Ä', 'À', '³'},
{'Õ', 'Í', '.', '³', '%', 'Í', 'Ô', '³'},
{'Ö', 'Ä', '.', 'º', '%', 'Ä', 'Ó', 'º'},
{'Û', 'Þ', 'Û', 'þ', 'Û', 'Û', 'Û', 'Ý'},
{'²', '±', '²', '±', '²', '±', '²', '±'}
};

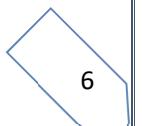
style%=6;
xsize+=2;
ysize+=2;

int xdel, ydel;

if(speed!=0)
{
    xdel = 100/speed;
    ydel = 200/speed;
}
else
{
    xdel=0;ydel=0;
}

gotoxy(xpos+1, ypos+1);
cout<<sym[style][0];
for(int i=(xpos+2);i<(xpos + xsize );i++){
    gotoxy(i, ypos + 1);
    cout<<sym[style][1];
    delay(xdel);
}
}

```



```

gotoxy((xpos + xsize ), (ypos +1));
cout<<sym[style][2];

for(int j=(ypos + 2); j<(ysize + ypos - 1);j++){
    gotoxy((xpos + xsize),j);
    cout<<sym[style][3];
    delay(ydel);
}

gotoxy((xpos + xsize   ), (ypos + ysize - 1 ));
cout<<sym[style][4];

for(i=(xpos + xsize - 1);i>(xpos + 1);i--){
    gotoxy(i,(ypos + ysize -1));
    cout<<sym[style][5];
    delay(xdel);
}

gotoxy(xpos+1 , ypos + ysize -1);
cout<<sym[style][6];

for(j=(ypos + ysize -2);j>(ypos + 1);j--){
    gotoxy(xpos + 1,j);
    cout<<sym[style][7];
    delay(ydel);
}

}

void header(int speed = 70)
{
    box(0,0,77,24,speed);
    gotoxy(28,1);
    puts(title);
}

```

```

}

void checkDir()
{
    struct stat buffer;

    if(stat("MAIN_DIR",&buffer)!=0)
    {mkdir("MAIN_DIR"); chdir("MAIN_DIR");
    char dir[3][25] = {"STD_DIR", "TCH_DIR","BUS_DIR"};
    for(int i=0; i<3; ++i) mkdir(dir[i]);
    }
    else chdir("MAIN_DIR");
}

void error(char a[50],char b[50],int apos,int bpos)
{
    _setcursortype(_NOCURSOR);
    box(19,8,40,10,0);
    for(int i = 21; i<=60; i++)
    {
        for( int j = 10; j<= 18; j++)
        {
            gotoxy(i,j);
            cout<<" ";
        }
    }
    gotoxy(37,10);    cout<<"WARNING!";
    gotoxy(apos,13);  cout<<a;
    gotoxy(bpos,15);  cout<<b;
    gotoxy(22,18);    cout<<"Press any key to continue.";
}

```

```

        getch();

    }

void changeDir(int a)
{
    char dir[3][25] = {"STD_DIR", "TCH_DIR","BUS_DIR"};
    char strDir[129];
    getcwd(strDir, 128);
    if(strstr(strDir,dir[a])!='\0')
        return;
    for(int i=0; i<3;++i)
        if( strstr(strDir,dir[i]) !='\0' ) chdir(..);

    chdir(dir[a]);
}

void addDatabase(int a, char admn[], char name[], char father[])
{
    changeDir(a);
    ofstream out;
    out.open("DB.txt",ios::app);
    out<<admn<<'\n'<<name<<' '<<father<<'\n';
    out.close();
}

int addOccupant(char filename[], char id[], char first[], char last[], char type)
{
    if(strcmpi(filename,"\\0")==0) return 0;
    changeDir(2);
}

```

```

ofstream put;
put.open(filename,ios::app);
put<<id<<' '<<first<<' '<<last<<' '<<type<<'\n';
put.close();
ifstream get;
char data[25];
get.open(filename);
get>>data>>data;
if(strcmpi(data,"Sharjah")==0)      return 250;
else if(strcmpi(data,"Dubai")==0)    return 350;
else if(strcmpi(data,"Ajman")==0)   return 260;
else if(strcmpi(data,"Dhaid")==0)    return 350;

get.close();
return 0;
}

```

```

class student
{
public:
char admn[15];
int grd;
char sec;
char name[50];
char fathersname[50];
char bus[5];
char transp;
char gender[10];
char dob[10];
char stream[10];
char subject[6][25];

```

```
    int fees;

    student();

};

student::student()
{
    grd=fees=0;
    transp=sec='\0';
    strcpy(admn,'\'0');
    strcpy(gender,'\'0');
    strcpy(bus,'\'0');
    strcpy(name,'\'0');
    strcpy(fathersname,'\'0');
    strcpy(dob,'\'0');

}

class teacher
{
public:
    char id[15];
    char fname[25];
    char lname[25];
    char mobile[15];
    char bus[5];
    char transp;
    char gender[10];
    char dob[10];
    char subject[25];
    char classt[5];
    char grd[4];
}
```

```

char sec[2];
char lvl[25];

teacher();

};

teacher::teacher()
{
    transp='\0';
    strcpy(grd,'0');
    strcpy(sec,'0');
    strcpy(id,'0');
    strcpy(gender,'0');
    strcpy(bus,'0');
    strcpy(fname,'0');
    strcpy(lname,'0');
    strcpy(mobile,'0');
    strcpy(dob,'0');
    strcpy(classt,'0');
    strcpy(subject,'0');
    strcpy(lvl,'0');
}

class transport
{
public:
    char no[15];
    char location[25];      //SHJ DUBAI OR AJMAN
    char driver[25];
    char dmobile[15];
    char conductor[25];
    char cmobile[15];
}

```

```

        transport();
    };

transport::transport()
{
    strcpy(no,'\0');
    strcpy(location,'\0');
    strcpy(driver,'\0');
    strcpy(dmobile,'\0');
    strcpy(conductor,'0');
    strcpy(cmobile,'0');

}

void viewBus()
{
    clrscr();
    char section[100]="#TRANSPORT RECORD VIEW FORM";
    char ch;
    int selection=0;
    header(30);
    struct stat buffer;

    changeDir(2);
    if(stat("DB.txt",&buffer)!=0) //Change to !=0
    {
        error("THERE ARE NO RECORDS IN DATABASE","ADD NEW RECORD
BEFORE PROCEEDING",25,25);
        return;
    }

    while(ch!=13)
    {
        clrscr();

```

```

header(0);

box(2, 4, 73, 19, 0, 2);

line(2,3,78,24,0,1);

gotoxy(6,5); puts("µCHOOSE ANY ONE OF THE FIELDSÆ");
gotoxy(4,3);
puts(section);

if(selection!=0 && ch==72)
    selection--;
if(selection!=2 && ch==80)
    selection++;

gotoxy(22, 11);   puts("Bus No: ");
gotoxy(22, 15);   puts("Driver Name: ");
gotoxy(22, 17);   puts("Conductor Name: ");

if(selection == 0)      box(19,9,37,2,0,1);
else if(selection == 1) box(19,13,37,4,0,1);
ch=getch();

}

ifstream get;
int found=0;
_setcursortype(_NORMALCURSOR);

char data[50]='\0';
switch(selection)
{
    case 0:
        char id[15]='\0';

```

```

gotoxy(30,11); gets(id);

if(strcmp(id,"\\0")==0) goto notfound;
strcat(id,".txt");

struct stat buff;
if(stat(id,&buff)==0) found=1;
if(!found)
{
    notfound:

    error("RECORD DOES NOT EXIST IN DATABASE","NO MATCH
        FOUND FOR INFO PROVIDED",24,25);
    return;
}
else   get.open(id);

break;

case 1:
char driver[25] , cond[25];
gotoxy(34,15); gets(driver);
gotoxy(33, 17); gets(cond);

if(strcmp(driver,"\\0")==0||strcmp(cond,"\\0")==0)
    goto notfound;

get.open("DB.txt");
int flag=0;

while(get.good())
{

```

```

get>>data>>data;

for(int j =0;j<strlen(data);++j)
    if(data[j] == '_') data[j]=' ';

if(strcmpi(data,driver)==0)
{
    get>>data;
    for(int j =0;j<strlen(data);++j)
        if(data[j] == '_') data[j]=' ';
    if(strcmpi(data,cond)== 0)
    {found=1; break; }
}

get>>data;
flag++;

}

get.close();

if(!found) goto notfound;
else
{
    get.open("DB.txt");
    for(int i=0; i<flag;++i)
        get>>data>>data>>data;
    get>>data;
    strcat(data,".txt");
    get.close();
    get.open(data);
}

break;

```

```

}

ch=0;
selection=1;

_setcursortype(_NOCURSOR);

transport bus;
get>>bus.no>>bus.location>>bus.driver>>bus.dmobile>>bus.conductor>>
bus.cmobile;

int max = 1,flag=0;

while(get.good())
{
    get>>data;
    if(get.eof()||max==3) break;
    else if(flag==0) max++;
    get>>data>>data>>data;
    flag++;
    if(flag>11) flag=0;
}

for(int j=0;j<strlen(bus.driver);++j)
    if(bus.driver[j]=='_') bus.driver[j]=' ';
for(j=0;j<strlen(bus.conductor);++j)
    if(bus.conductor[j]=='_') bus.conductor[j]=' ';

while(ch!=13)
{
    clrscr();
}

```

```

header(0);

box(2, 4, 73, 19, 0, 4);
line(2,3,78,24,0,1);
gotoxy(4,3);
puts("µTRANSPORT RECORD VIEWÆ");
if(max!=1)
{
    if(selection!=1 && ch==75)    selection--;
    if(selection!=max && ch==77)  selection++;
}
gotoxy(4,22); cout<<"PRESS ENTER TO GO BACK";
gotoxy(4,23); cout<<"ARROW KEY TO CHANGE PAGE";
gotoxy(62, 24); printf("µPAGE %i OF %iÆ",selection,max);

if(selection==1)      goto page1;
else if(selection==2)  goto page2;
else if(selection==3)  goto page3;

page1:
gotoxy(35,7);    puts("BUS DETAILS");
gotoxy(5, 11);   printf("Bus No: %s",bus.no);
gotoxy(5, 15);   printf("Driver Name: %s ",bus.driver);

gotoxy(40, 15);  printf("Contact No: %s",bus.dmobile);

gotoxy(5, 19);   printf("Conductor Name: %s",bus.conductor);

gotoxy(40, 19);  printf("Contact No: %s",bus.cmobile);

ch=getch();
continue;

```

```

page2:
int p;
page3:
int ypos=11; int i=0;
char name[25],type[10];
gotoxy(33,6);    puts("OCCUPANT DETAILS");
gotoxy(5,6);    cout<<"(ONLY FIRST 22)";
gotoxy(7,8);    puts("SI No");
gotoxy(15,8);   puts("ID No");
gotoxy(25,8);   puts("Name");
gotoxy(60,8);   puts("Status");
line(5,9,75,1,0,2);

if(selection!=2) goto skip;
get.clear();
get.seekg(0,ios::beg);
for( ; i<7;++i) get>>data;
p=1;
skip:
flag=0;
while(get.good())
{
    gotoxy(7,ypos); cout<<p;
    gotoxy(15,ypos); puts(data);
    get>>data; strcpy(name,data);
    get>>data; strcat(data,name);
    get>>type;
    if(strcmpi(type,"S")==0) strcat(type,"tudent");
    else strcat(type,"eacher");
    gotoxy(25,ypos); puts(name);
    gotoxy(60,ypos); puts(type);
}

```

```

        get>>data;
        ypos+=1;
        flag++;
        p++;
        if(flag==11)      break;
    }

    ch=getch();
    continue;
}

get.close();

}

void viewTeacher()
{
    clrscr();
    char section[100]="#TEACHER RECORD VIEW FORM#";
    char ch;
    int selection=0;
    header(30);
    struct stat buffer;

    changeDir(1);
    if(stat("DB.txt",&buffer)!=0) //Change to !=0
    {
        error("THERE ARE NO RECORDS IN DATABASE","ADD NEW RECORD
BEFORE PROCEEDING",25,25);
        return;
    }

    while(ch!=13)

```

```

{
    clrscr();
    header(0);
    box(2, 4, 73, 19, 0, 2);
    line(2,3,78,24,0,1);

    gotoxy(6,5); puts("CHOOSE ANY ONE OF THE FIELDS");
    gotoxy(4,3);
    puts(section);

    //Adm no or Name or Class

    gotoxy(22,11);    puts("ID No: ");
    gotoxy(22, 15);   puts("First Name: ");
    gotoxy(22, 17);   puts("Last Name: ");

    if(selection!=0 && ch==72)
        selection--;
    if(selection!=1 && ch==80)
        selection++;
    if(selection == 0)      box(19,9,37,2,0,1);
    else if(selection == 1)  box(19,13,37,4,0,1);

    ch=getch();
}

ifstream get;
int found=0;
_setcursortype(_NORMALCURSOR);

switch(selection)

```

```

{

    case 0:

        char id[15] = "\0";

        gotoxy(29,11); gets(id);

        if(strcmp(id, "\0") == 0) goto notfound;

        strcat(id, ".txt");




        struct stat buff;

        if(stat(id, &buff) == 0) found = 1;

        if(!found)

        {

            notfound:

            error("RECORD DOES NOT EXIST IN DATABASE", "NO MATCH FOUND FOR INFO PROVIDED", 24, 25);

            return;

        }

        else get.open(id);





        break;





    case 1:

        char firstname[25], lastname[25];

        gotoxy(34,15); gets(firstname);

        gotoxy(33, 17); gets(lastname);





        if(strcmp(firstname, "\0") == 0 || strcmp(lastname, "\0") == 0)

            goto notfound;

```

```

char data[50]= "\0";

get.open("DB.txt");
int flag=0;

while(get.good())
{
    get>>data>>data;

    for(int j =0;j<strlen(data);++j)
        if(data[j] == '_') data[j]=' ';

    if(strcmpi(data,firstname)==0)
    {
        get>>data;
        for(int j =0;j<strlen(data);++j)
            if(data[j] == '_') data[j]=' ';
        if(strcmpi(data,lastname)== 0)
        {found=1; break; }

    }
    get>>data;
    flag++;

}

get.close();

if(!found) goto notfound;
else
{
    get.open("DB.txt");
    for(int i=0; i<flag;++i)

```

```

        get>>data>>data>>data;
        get>>data;
        strcat(data,".txt");
        get.close();
        get.open(data);
    }

    break;
}

ch=0;
selection=1;

_setcursortype(_NOCURSOR);

teacher tch;
get>>tch.id>>tch.fname>>tch.lname>>tch.classt>>tch.mobile
>>tch.bus>>tch.gender>>tch.dob>>tch.lvl>>tch.subject;
get.close();

for(int j=0;j<strlen(tch.subject);++j)
    if(tch.subject[j]=='_') tch.subject[j]=' ';
for(j=0;j<strlen(tch.lvl);++j)
    if(tch.lvl[j]=='_')      tch.lvl[j]=' ';

while(ch!=13)
{
    clrscr();
    header(0);
    box(2, 4, 73, 19, 0, 4);
    line(2,3,78,24,0,1);
}

```

```

gotoxy(4,3);
puts("μTEACHER RECORD VIEW");
if((int)ch == 75 && selection==2) selection = 1;
if((int)ch == 77 && selection==1) selection = 2;

gotoxy(4,22); cout<<"PRESS ENTER TO GO BACK";
gotoxy(4,23); cout<<"ARROW KEY TO CHANGE PAGE";
gotoxy(62, 24); printf("μPAGE %i OF 2",selection);

if(selection==1) goto page1;
else if(selection==2) goto page2;

page1:
gotoxy(33,7); puts("PERSONAL DETAILS");
gotoxy(5, 11); printf("ID No: %s",tch.id);
gotoxy(40, 11); printf("Name: %s %s",tch.fname,tch.lname);

gotoxy(5, 15); printf("Gender: %s",tch.gender);
gotoxy(40, 15); printf("Mobile No: %s",tch.mobile);

gotoxy(5, 19); printf("Bus No: %s",tch.bus);
gotoxy(40, 19); printf("Date of Birth: %s",tch.dob);

ch=getch();
continue;

page2:
int pos=14;
gotoxy(35,7); puts("JOB DETAILS");
if(strcmpi(tch.classt,"N/A")!=0)
{
    gotoxy(24,13);
    printf("Class Teacher of %s", tch.classt);
}

```

```

        pos++;
    }

    box(14,9,50,10,0,2);

    gotoxy(24,pos); printf("%s Teacher",tch.lvl); pos+=2;
    gotoxy(24,pos); printf("Subject: %s",tch.subject );
    ch=getch();
    continue;

}

}

void viewStudent()
{
    clrscr();
    char section[100]="#STUDENT RECORD VIEW FORM#";
    char ch;
    int selection=0;
    header(30);
    struct stat buffer;

    changeDir(0);
    if(stat("DB.txt",&buffer)!=0) //Change to !=0
    {
        error("THERE ARE NO RECORDS IN DATABASE","ADD NEW RECORD
BEFORE PROCEEDING",25,25);
        return;
    }

    while(ch!=13)

```

```

{
    clrscr();
    header(0);
    box(2, 4, 73, 19, 0, 2);
    line(2,3,78,24,0,1);

    gotoxy(6,5); puts("CHOOSE ANY ONE OF THE FIELDS");
    gotoxy(4,3);
    puts(section);

    //Adm no or Name or Class

    gotoxy(22,11);    puts("Admn No: ");
    gotoxy(22, 15);   puts("Name: ");
    gotoxy(22, 17);   puts("Father's Name: ");

    if(selection!=0 && ch==72)
        selection--;
    if(selection!=1 && ch==80)
        selection++;
    if(selection == 0)      box(19,9,37,2,0,1);
    else if(selection == 1) box(19,13,37,4,0,1);

    ch=getch();
}

ifstream get;
int found=0;
_setcursortype(_NORMALCURSOR);

switch(selection)

```

```

{

    case 0:

        char admn[15] = "\0";

        gotoxy(31,11); gets(admn);
        if(strcmp(admn, "\0") == 0) goto notfound;
        strcat(admn, ".txt");

        struct stat buff;
        if(stat(admn, &buff) == 0) found = 1;
        if(!found)
        {
            notfound:

                error("RECORD DOES NOT EXIST IN DATABASE", "NO MATCH
FNEWFOUND FOR INFO PROVIDED", 24, 25);
                return;
        }
        else
            get.open(admn);

        break;

    case 1:

        char name[25], father[25];
        gotoxy(28,15); gets(name);
        gotoxy(37, 17); gets(father);

        if(strcmp(name, "\0") == 0 || strcmp(father, "\0") == 0) goto
notfound;
}

```

```

char data[50] = "\0";

get.open("DB.txt");
int flag=0; found=0;

while(get.good())
{
    get>>data>>data;

    for(int j = 0; j < strlen(data); ++j)
        if(data[j] == '_') data[j] = ' ';

    if(strcmpi(data, name) == 0)
    {
        get>>data;
        for(int j = 0; j < strlen(data); ++j)
            if(data[j] == '_') data[j] = ' ';
        if(strcmpi(data, father) == 0)
        {found=1; break; }
    }

    get>>data;
    flag++;
}

get.close();

if(!found) goto notfound;
else
{
    get.open("DB.txt");
    for(int i=0; i<flag; ++i)

```

```

        get>>data>>data>>data;
        get>>data;
        strcat(data,".txt");
        get.close();
        get.open(data);

    }

    break;
}

//Data viewing area
ch=0;
selection=1;

_setcursortype(_NOCURSOR);

student st;
get>>st.admn>>st.name>>st.fathersname>>st.grd>>st.sec>>st.bus>>
st.gender>>st.dob>>st.stream>>st.fees;
for(int j=0;j<6;++j) get>>st.subject[j];
for(j=0;j<6;++j)
{
    for(int k=0;k<strlen(st.subject[j]);++k)
        if(st.subject[j][k]=='_')      st.subject[j][k]=' ';
}

for(j=0;j<strlen(st.name);++j)
    if(st.name[j]=='_')      st.name[j]=' ';
for(j=0;j<strlen(st.fathersname);++j)
    if(st.fathersname[j]=='_')    st.fathersname[j]=' ';

```

```

get.close();

while(ch!=13)
{
    clrscr();
    header(0);
    box(2, 4, 73, 19, 0, 4);
    line(2,3,78,24,0,1);
    gotoxy(4,3);
    puts("µSTUDENT RECORD VIEWÆ");
    if((int)ch == 75 && selection==2) selection = 1;
    if((int)ch == 77 && selection==1) selection = 2;

    gotoxy(4,22); cout<<"PRESS ENTER TO GO BACK";
    gotoxy(4,23); cout<<"ARROW KEY TO CHANGE PAGE";
    gotoxy(62, 24); printf("µPAGE %i OF 2Æ",selection);

    if(selection==1) goto page1;
    else if(selection==2) goto page2;

page1:
    gotoxy(33,7);      puts("PERSONAL DETAILS");
    gotoxy(5, 9);      printf("Admn No: %s",st.admn);
    gotoxy(40, 9);     printf("Class: %i %c",st.grd,st.sec);

    gotoxy(5, 13);     printf("Student Name: %s",st.name);

    gotoxy(40, 13);    printf("Father Name: %s",st.fathersname);

    gotoxy(5, 17);     printf("Gender: %s",st.gender);

    gotoxy(40, 17);    printf("Date of Birth: %s",st.dob);
    gotoxy(5, 20);     printf("Bus No: %s",st.bus);

```

```

        ch=getch();
        continue;

    page2:
    if(strcmp(st.stream,"N/A")!=0)
    {
        gotoxy(5,7);      printf("%s Stream", st.stream);
    }
    gotoxy(5,8);      printf("Fees: %i /month", st.fees);
    box(14,9,50,10,0,2);
    gotoxy(29,10);   puts("STUDENT'S SUBJECT LIST");
    int xpos=17,ypos=12;
    for(int i=0; i<6;++i)
    {

        if((i%2)==1) xpos+=28; else if(i!=0) {xpos=17; ypos+=3;}
        gotoxy(xpos,ypos);   puts(st.subject[i]);
    }
    ch=getch();
    continue;

}

void addStudent()
{
    clrscr();
    header(30);
    char section[100] = "STUDENT RECORD ADDITION FORM";
    box(2, 4, 73, 19, 0, 2);
}

```

```

line(2,3,78,24,0,1);

gotoxy(4,3);
puts(section);

student st;

gotoxy(5, 8);    puts("Admn No: ");
gotoxy(40, 8);   puts("Grade: ");
gotoxy(60, 8);   puts("Section: ");
gotoxy(5, 12);   puts("Name: ");
gotoxy(40, 12);  puts("Father's Name: ");
gotoxy(5, 16);   puts("Gender (M/F): ");
gotoxy(40, 16);  puts("Date of Birth: DDMMYYYY");
gotoxy(5, 20);   puts("School Transport?(Y/N): ");

_setcursortype(_NORMALCURSOR);

gotoxy(14,8); gets(st.admn);
//Checking dup value
struct stat buffer;

char filename[25]; strcpy(filename,st.admn);
strcat(filename,".txt");

changeDir(0);

if(stat(filename,&buffer)==0 || strcmp(st.admn,"\\0")==0)
{
    error("INCORRECT / EXISTING ID","PLEASE CHECK THE VALUE AND
TRY AGAIN",29,23);
    return;
}

gotoxy(47, 8);    cin>>st.grd;
gotoxy(69, 8);    cin>>st.sec;      st.sec = toupper(st.sec);

```

```

        gotoxy(11,12);    gets(st.name);
        gotoxy(55,12);    gets(st.fathersname);
        gotoxy(19,16);    cin>>st.gender;
        gotoxy(55,16);    gets(st.dob);

        gotoxy(29,20);    cin>>st.transp;

        char busname[25] = "\0";
        if(st.transp=='Y'||st.transp=='y')
        {

            gotoxy(40,20);    puts("Bus No: ");
            gotoxy(48,20);    gets(st.bus);

            strcpy(busname,st.bus); strcat(busname,".txt");
            changeDir(2);
            struct stat buffer1;

            if(stat(busname,&buffer1)!=0)
            {
                error("BUS NOT FOUND IN DATABASE","ADD BUS TO DATABASE
BEFORE PROCEEDING",29,23);
                return;
            }

        }
        else strcpy(st.bus,"N/A");

        if(st.gender[0]!='m'||st.gender[0]!='M'||st.gender[0]!='F'||st.gender[0]!='f')
            strcpy(st.gender,"M");

```

```

st.gender[0] = toupper(st.gender[0]);
if(st.gender[0]=='M')    strcat(st.gender,"ale");
else strcat(st.gender,"emale");

if(strlen(st.dob)<8)    strcpy(st.dob,"N/A");

_setcursortype(_NOCURSOR);
strcpy(st.subject[5], "Arabic");

if(st.grd==12 || st.grd==11)
{
    int selection=0;
    char ch;

    while(ch!=13)
    {

        clrscr();
        header(0);
        box(2, 4, 73, 19, 0, 2);
        line(2,3,78,24,0,1);
        gotoxy(4,3);
        puts(section);

        gotoxy(15,8);    puts("Enter stream: ");
        gotoxy(15,11);   puts("Science");
        gotoxy(15,14);   puts("Commerce");

        if((int)ch == 72 && selection==1) selection = 0;
        if((int)ch == 80 && selection==0) selection = 1;
        if(selection == 0) box(11,9,15,2,0);
    }
}

```

```

        else if(selection == 1) box(11,12,15,2,0);
        ch=getch();
    }

    if(selection==0)      strcpy(st.stream,"Science");
    else if(selection == 1) strcpy(st.stream,"Commerce");



    if(strcmpi(st.stream,"science")==0)
    {
        strcpy(st.subject[0], "Physics");
        strcpy(st.subject[1], "Chemistry");
        strcpy(st.subject[2], "English");
    }

    if(strcmpi(st.stream,"commerce")==0)
    {
        strcpy(st.subject[0], "Business_Studies");
        strcpy(st.subject[1], "Accountancy");
        strcpy(st.subject[2], "English");
        strcpy(st.subject[3], "Marketing");
    }

    char subopt[3][2][25] =
    {
        {"Comp Science", "Biology"},

        {"Maths", "Info"},

        {"Economics", "Info"}
    };

    int subsel = 0;

```

```

int opt;

if(strcmpi(st.stream,"science")==0)      opt=0;
else if(strcmpi(st.stream,"commerce")==0) opt=2;

ch=0;

do
{
    while(ch!=13)

    {
        clrscr();
        header(0);
        box(2, 4, 73, 19, 0, 2);
        line(2,3,78,24,0,1);
        gotoxy(4,3);
        puts(section);
        gotoxy(15,8);    puts("Enter stream: ");
        gotoxy(15,11);   puts("Science");
        gotoxy(15,14);   puts("Commerce");

        if(strcmpi(st.stream,"science")==0)
            box(11,9,15,2,0);
        else if(strcmpi(st.stream,"commerce")==0)
            box(11,12,15,2,0);

        gotoxy(35,8);    puts("Subject: ");
        gotoxy(35,11);   puts(subopt[opt][0]);
        gotoxy(35,14);   puts(subopt[opt][1]);
        if((int)ch == 72 && subsel==1) subsel = 0;
        if((int)ch == 80 && subsel==0) subsel = 1;
        if(subsel == 0)      box(31,9,15,2,0);
    }
}

```

```

        else if(subsel == 1)    box(31,12,15,2,0);
        ch=getch();
    }

if(subsel == 0 && strcmpi(st.stream,"science")==0)
{
    strcpy(st.subject[3],"Computer_Science");
    strcpy(st.subject[4],"Maths");
    break;
}

else if(strcmpi(st.stream,"commerce")==0)
{
    strcpy(st.subject[4],subopt[opt][subsel]);
    break;
}

else
{
    int asel = 0;
    ch=0;

    while(ch!=13)
    {
        clrscr();
        header(0);
        box(2, 4, 73, 19, 0, 2);
        line(2,3,78,24,0,1);
        gotoxy(4,3);
        puts(section);
    }
}

```

```
    gotoxy(15,8);      puts("Enter stream: ");
    gotoxy(15,11);    puts("Science");
    gotoxy(15,14);    puts("Commerce");
```

```
    if(strcmpi(st.stream,"science")==0)
        box(11,9,15,2,0);
    else if(strcmpi(st.stream,"commerce")==0)
        box(11,12,15,2,0);
```

```
    gotoxy(35,8);      puts("Subject: ");
    gotoxy(35,11);    puts(subopt[opt][0]);
    gotoxy(35,14);    puts(subopt[opt][1]);
```

```
    if(subsel == 0)      box(31,9,15,2,0);
    else if(subsel == 1)  box(31,12,15,2,0);
```

```
    gotoxy(55,8);      puts("Elected Subject: ");
    gotoxy(55,11);    puts(subopt[1][0]);
    gotoxy(55,14);    puts(subopt[1][1]);
```

```
    if((int)ch == 72 && asel==1) asel = 0;
    if((int)ch == 80 && asel==0) asel = 1;
```

```
    if(asel == 0)      box(51,9,15,2,0);
    else if(asel == 1) box(51,12,15,2,0);
```

```
    ch=getch();
```

```
}
```

```
strcpy(st.subject[3],subopt[opt][subsel]);
strcpy(st.subject[4], subopt[1][asel]);
```

```

        clrscr();
        break;
    }
} while(0);
}

else
{
    char lsubs[5][25]={"Science", "Maths", "English",
"Second_Language", "Social_Studies"};
    for(int i=0;i<5;++i)
        strcpy(st.subject[i],lsubs[i]);
    strcpy(st.stream,"N/A");
}

if(st.grd>=10)          st.fees+=750;
else if(st.grd>=7)       st.fees+=585;
else                      st.fees+=510;

for(int j =0;j<strlen(st.name);++j)
{
    if(j==0) st.name[0]=toupper(st.name[0]);

    if(st.name[j] == ' ')
    {
        st.name[j]='_'; st.name[j+1]= toupper(st.name[j]);
    }
}

for(j =0;j<strlen(st.fathersname);++j)
{
    if(j==0) st.fathersname[0]=toupper(st.fathersname[0]);

    if(st.fathersname[j] == ' ')
    {
}

```

```

        st.fathersname[j]='_';
        st.fathersname[j++] = toupper(st.fathersname[j]);
    }

}

st.fees+=addOccupant(busname,st.admn,st.name,st.fathersname,'S');

addDatabase(0,st.admn,st.name,st.fathersname);

//File handling starts

ofstream put;

changeDir(0); //change directory

put.open(filename);

put<<st.admn<<'\n'<<st.name<<'\t'<<st.fathersname<<'\n'<<st.grd<<'\t'
'<<st.sec<<'\n'<<st.bus<<'\n'<<st.gender<<'\n'<<st.dob<<'\n'<<st.str
eam<<'\n'<<st.fees;

for(int i = 0; i<6; ++i){put<<'\n'<<st.subject[i];}

put.close();

//File handling ends

clrscr();

header(0);

box(2, 4, 73, 19, 0, 2);

line(2,3,78,24,0,1);

gotoxy(4,3);

puts(section);

gotoxy(28,13); puts("Record successfully added!");

getch();

}

void remRcrd(char type[])
{
    clrscr();
}

```

```

header(30);

char section[100] = "μ";
strcat(section, type);
strcat(section, " RECORD DELETION FORMA");
box(2, 4, 73, 19, 0, 2);
line(2, 3, 78, 24, 0, 1);
gotoxy(4, 3);
puts(section);
if(strcmpi(type, "Student") == 0 || strcmpi(type, "Teacher") == 0)
{
    gotoxy(22, 15);    puts("ID No: ");
    box(19, 13, 37, 2, 0, 1);

    if(strcmpi(type, "Student") == 0) changeDir(0);
    else                                changeDir(1);
    struct stat buffer;
    if(stat("DB.txt", &buffer) != 0) //Change to !=0
    {
        error("THERE ARE NO RECORDS IN DATABASE", "ADD NEW RECORD
BEFORE PROCEEDING", 25, 25);
        return;
    }
    ifstream get;
    int found = 0;
    _setcursortype(_NORMALCURSOR);

    char id[15] = "\0";

    gotoxy(29, 15); gets(id);
    if(strcmp(id, "\0") == 0) goto notfound;
    strcat(id, ".txt");

```

```

    struct stat buff;
    if(stat(id,&buff)==0) found=1;
    if(!found)
    {
        error("RECORD DOES NOT EXIST IN DATABASE","NO MATCH
        FOUND FOR INFO PROVIDED",24,25);
        return;
    }
    else get.open(id);
    char data[15],search[10],file[10]; get>>search;
    for(int i=0;i<4;++i)    get>>data;
    get>>file;
    get.close();
    remove(id);
    //remove from DB
    ofstream put;
    put.open("temp.txt");
    get.open("DB.txt");
    i=0;

    while(get.good())
    {
        get>>data;
        if(get.eof())      break;
        if(strcmpi(search,data)==0)
            get>>data>>data;
        else
        {
            put<<data<<'\n';
            get>>data;  put<<data<<'\t';
            get>>data;  put<<data;
        }
    }

```

```

        ++i;

    }

    get.close();

    put.close();

    remove("DB.txt");

    rename("temp.txt","db.txt");



    if(i==1)      remove("DB.txt");



    if(strcmp(file,"N/A")!=0)
    {

        //removing from bus

        changeDir(2);

        strcat(file,".txt");

        get.open(file);

        char first[25],last[25],type;

        i=0;

        while(get.good())
        {

            put.open("temp.txt",ios::app);

            get>>data;

            if(get.eof())      break;

            if(strcmpl(search,data)==0 && i>5)

                get>>data>>data>>data>>data;

            if(i<=5) put<<data<<'\n';

            put.close();

            if(i>5)

            {

                get>>first>>last>>type;





addOccupant("temp.txt",data,first,last,type);

        }
    }

```

```

        ++i;
    }
    remove(file);
    rename("temp.txt",file);
    //removed
}
}

else if(strcmpi(type,"Transport")==0)
{
    gotoxy(22,15);    puts("Bus No: ");
    box(19,13,37,2,0,1);

    changeDir(2);
    struct stat buffer;
    if(stat("DB.txt",&buffer)!=0) //Change to !=0
    {
        notfound:
        error("THERE ARE NO RECORDS IN DATABASE","ADD NEW RECORD
BEFORE PROCEEDING",25,25);
        return;
    }
    ifstream get;
    int found=0;
    _setcursortype(_NORMALCURSOR);

    char id[15]='\0';

    gotoxy(30,15); gets(id);
    if(strcmp(id,"'\0')==0) goto notfound;
    strcat(id,".txt");
    struct stat buff;
    if(stat(id,&buff)==0) found=1;
}

```

```

if(!found)
{
    error("RECORD DOES NOT EXIST IN DATABASE","NO MATCH
    FOUND FOR INFO PROVIDED",24,25);

    return;
}

else get.open(id);

char data[15],search[10],file[10]; get>>search;
for(int i=0;i<5;++i) get>>data;

ofstream put;
ifstream pget;
while(get.good())
{
    get>>data;
    if(get.eof()) break;
    strcpy(file,data); strcat(file,".txt");
    get>>data>>data>>data;
    if(strcmpi(data,"S")==0) changeDir(0);
    else changeDir(1);
    pget.open(file);
    put.open("temp.txt");
    char sym[2]={'\n','\t'};
    for(int j=0;j<5;++j)
    {
        pget>>data; put<<data<<sym[j%2];
    }
    put<<"N/A"<<'\
'; pget>>data;
    while(pget.good())
    {
        pget>>data; put<<data<<'\
';
    }
}

```

```

    }

    pget.close();
    put.close();
    remove(file);
    rename("temp.txt",file);
    changeDir(2);

}

get.close();
remove(id);
//remove from DB
put.open("temp.txt");
get.open("DB.txt");
i=0;

while(get.good())
{
    get>>data;
    if(get.eof())      break;
    if(strcmpi(search,data)==0)
        get>>data>>data;
    else
    {
        put<<data<<'\n';
        get>>data;  put<<data<<'\t';
        get>>data;  put<<data;
    }
    ++i;
}

get.close();
put.close();
remove("DB.txt");
rename("temp.txt","db.txt");

```

```

        if(i==1)      remove("DB.txt");
    }

    clrscr();
    header(0);
    box(2, 4, 73, 19, 0, 2);
    line(2,3,78,24,0,1);
    gotoxy(4,3);
    puts(section);
    gotoxy(28,13); puts("Record successfully removed!");
    getch();
}

void addBus()
{
    clrscr();
    header(30);
    char section[100] = "TRANSPORT RECORD ADDITION FORM";
    box(2, 4, 73, 19, 0, 2);
    line(2,3,78,24,0,1);
    gotoxy(4,3);
    puts(section);

    transport bus;
    gotoxy(35,7);    puts("BUS DETAILS");
    gotoxy(5, 10);   puts("Bus No: ");
    gotoxy(5, 14);   puts("Driver Name: ");
    gotoxy(44, 14);  puts("Mobile No: ");
    gotoxy(5, 18);   puts("Conductor Name: ");
}

```

```

gotoxy(44,18);    puts("Mobile No: ");

_setcursortype(_NORMALCURSOR);
gotoxy(13,10); gets(bus.no);
//Checking dup value
struct stat buffer;

char filename[25]; strcpy(filename,bus.no); strcat(filename,".txt");
changeDir(2);
if(stat(filename,&buffer)==0 || strcmp(bus.no,"\\0")==0)
{
    error("INCORRECT / EXISTING ID","PLEASE CHECK THE VALUE AND
TRY AGAIN",29,23);
    return;
}

gotoxy(18,14); gets(bus.driver);
gotoxy(55,14); gets(bus.dmobile);
gotoxy(21,18); gets(bus.conductor);
gotoxy(55,18); gets(bus.cmobile);

_setcursortype(_NOCURSOR);

int selection=0;
char ch;
char loc[4][25]={"Sharjah","Dubai","Ajman","Dhaid"};
while(ch!=13)
{

    clrscr();
    header(0);
    box(2, 4, 73, 19, 0, 2);
}

```

```

line(2,3,78,24,0,1);

gotoxy(4,3);
puts(section);

if(selection!=0 && ch==72)    selection--;
if(selection!=3 && ch==80)    selection++;
gotoxy(35,8);    puts("Location: ");
int ypos=11;
for(int i=0;i<4;++i)
{
    if(i!=0) ypos+=3;
    gotoxy(33,ypos); puts(loc[i]);
    if(i==selection) box(30,ypos-2,18,2,0);
}
ch=getch();
}

strcpy(bus.location,loc[selection]);

for(int j =0;j<strlen(bus.driver);++j)
{   if(j==0) bus.driver[0]=toupper(bus.driver[0]);

    if(bus.driver[j] == ' ')
    {
        bus.driver[j]='_'; bus.driver[j+1]=
toupper(bus.driver[j]);
    }
}

for(j =0;j<strlen(bus.conductor);++j)
{   if(j==0) bus.conductor[0]=toupper(bus.conductor[0]);

    if(bus.conductor[j] == ' ')
    {

```

```

        bus.conductor[j]='_';
        bus.conductor[j++] = toupper(bus.conductor[j]);
    }
}

addDatabase(2,bus.no,bus.driver,bus.conductor);

//File handling starts
ofstream put;
changeDir(2); //change directory
put.open(filename);
put<<bus.no<<'\n'<<bus.location<<'\n'<<bus.driver<<'\n'<<bus.dmobile
<<'\n'<<bus.conductor<<'\n'<<bus.cmobile<<'\n';
put.close();

//File handling ends
clrscr();
header(0);
box(2, 4, 73, 19, 0, 2);
line(2,3,78,24,0,1);
gotoxy(4,3);
puts(section);
gotoxy(28,13); puts("Record successfully added!");
getch();

}

void addTeacher()
{
    clrscr();
    header(30);
    char section[100]="#TEACHER RECORD ADDITION FORM#";
    box(2, 4, 73, 19, 0, 2);
}

```

```

line(2,3,78,24,0,1);

gotoxy(4,3);
puts(section);

teacher tch;

gotoxy(5, 8);      puts("Staff ID: ");
gotoxy(44, 8);    puts("Mobile No: ");
gotoxy(5, 12);    puts("First Name: ");
gotoxy(44, 12);   puts("Last Name: ");
gotoxy(5, 16);    puts("Date of Birth: DDMMYYYY");
gotoxy(44,16);   puts("Gender (M/F): ");
gotoxy(5, 20);    puts("School Transport?(Y/N): ");
gotoxy(44, 20);   puts("Class Teacher?(Y/N): ");

_setcursortype(_NORMALCURSOR);
gotoxy(15,8); gets(tch.id);
//Checking dup value
struct stat buffer;

char filename[25]; strcpy(filename,tch.id); strcat(filename,".txt");
changeDir(1);
if(stat(filename,&buffer)==0 || strcmp(tch.id,"\\0")==0)
{
    error("INCORRECT / EXISTING ID","PLEASE CHECK THE VALUE AND
TRY AGAIN",29,23);
    return;
}

gotoxy(55,8);    gets(tch.mobile);
gotoxy(17,12);   gets(tch.fname);
gotoxy(55,12);   gets(tch.lname);

```

```

gotoxy(20,16);    gets(tch.dob);
gotoxy(58,16);   cin>>tch.gender;

gotoxy(29,20);   cin>>tch.transp;
char busname[25] = "\0";
if(tch.transp=='Y'||tch.transp=='y')
{
    gotoxy(21,22);   puts("Bus No: ");
    gotoxy(29,22);   gets(tch.bus);

    strcpy(busname,tch.bus); strcat(busname,".txt");
    changeDir(2);
    struct stat buffer1;

    if(stat(busname,&buffer1)!=0)
    {
        error("BUS NOT FOUND IN DATABASE","ADD BUS TO DATABASE
BEFORE PROCEEDING",29,23);
        return;
    }
    else strcpy(tch.bus,"N/A");

    gotoxy(65, 20); cin>>tch.classt;
    if(tch.classt[0]=='Y' || tch.classt[0]=='y')
    {
        gotoxy(40, 22);   puts("Grade: ");
        gotoxy(60, 22);   puts("Section: ");
        gotoxy(47, 22);   gets(tch.grd);
        gotoxy(69, 22);   cin>>tch.sec; tch.sec[0] =
toupper(tch.sec[0]);
        strcpy(tch.classt,tch.grd);      strcat(tch.classt,tch.sec);
    }
}

```

```

else strcpy(tch.classt,"N/A");

if(tch.gender[0]!='m' || tch.gender[0]!='M' || tch.gender[0]!='F' || tch.gender[0]!='f')
    strcpy(tch.gender,"M");
tch.gender[0] = toupper(tch.gender[0]);
if(tch.gender[0]=='M') strcat(tch.gender,"ale");
else strcat(tch.gender,"emale");

if(strlen(tch.dob)<8) strcpy(tch.dob,"N/A");

_setcursortype(_NOCURSOR);

int selection=0;
char ch;

char level[3][25]=
{"Primary","Secondary","Higher Secondary"};
while(ch!=13)
{

clrscr();
header(0);
box(2, 4, 73, 19, 0, 2);
line(2,3,78,24,0,1);
gotoxy(4,3);
puts(section);

if(selection!=0 && ch==72) selection--;
if(selection!=2 && ch==80) selection++;
gotoxy(33,8); puts("Teaching Level: ");
int ypos=11;

```

```

        for(int i=0;i<3;++i)
        {
            if(i!=0) ypos+=3;
            gotoxy(33,ypos); puts(level[i]);
            if(i==selection) box(30,ypos-2,18,2,0);
        }
        ch=getch();
    }

char subjects[3][15][25]=
{
    {"Science", "Maths", "English", "Second Language", "Social Studies",
     "Arabic"},

    {"Physics", "Chemistry", "Biology", "Maths", "English", "Second
     Language", "Social Studies", "Arabic"},

    {"Physics", "Chemistry", "Biology", "Maths", "English",
     "Computer Science", "Info", "Economics", "Business Studies",
     "Accountancy", "Marketing", "Arabic"}
};

int prevsel=selection;
int max,ypos,ysize,y;
ch=0;

if(selection==0)      {max=6; y=11; ysize=3;}
else if(selection==1) {max=8;      y=10; ysize=3;}
else if(selection==2) {max=12;y=10; ysize=2; }

selection=0;
while(ch!=13)
{
    clrscr();
    header(0);
}

```

```

        box(2, 4, 73, 19, 0, 2);
        line(2,3,78,24,0,1);
        gotoxy(4,3);
        puts(section);
        gotoxy(4,23); cout<<"USE ARROW KEY TO NAVIGATE";
        gotoxy(6,6); printf("Teaching Level: %s",level[prevsel]);
        if(selection!=0 && ch==72)    selection--;
        if(selection!=max-1 && ch==80)    selection++;
        int xpos=17;
        for(int i=0; i<max;++i)
        {
            if(i==0) ypos=y;
            else if(i==max/2) {xpos+=28; ypos = y;} else if(i!=0)
                ypos+=ysize;
            gotoxy(xpos,ypos);    puts(subjects[prevsel][i]);
            if(i==selection)    box(xpos-3,ypos-2,17,2,0);
        }

        ch=getch();

    }
    strcpy(tch.lvl,level[prevsel]);
    strcpy(tch.subject,subjects[prevsel][selection]);

    for(int j =0;j<strlen(tch.subject);++j)
        if(tch.subject[j] == ' ')    tch.subject[j]='_';

    for(j =0;j<strlen(tch.lvl);++j)
        if(tch.lvl[j] == ' ')    tch.lvl[j]='_';

    tch.fname[0]=toupper(tch.fname[0]);
    tch.lname[0]=toupper(tch.lname[0]);

```

```

addDatabase(1,tch.id,tch.fname,tch.lname);
addOccupant(busname,tch.id,tch.fname,tch.lname,'T');

//File handling starts
ofstream put;
changeDir(1); //change directory
put.open(filename);
put<<tch.id<<'\n'<<tch.fname<<'\t'<<tch.lname<<'\n'<<tch.classt<<'\n'
'<<tch.mobile<<'\n'<<tch.bus<<'\n'<<tch.gender<<'\n'<<tch.dob<<'\n'<
<tch.lvl<<'\n'<<tch.subject;
put.close();

//File handling ends
clrscr();
header(0);
box(2, 4, 73, 19, 0, 2);
line(2,3,78,24,0,1);
gotoxy(4,3);
puts(section);
gotoxy(28,13); puts("Record successfully added!");
getch();

}

void menuHandler(char type[ ]='\\0')
{
    _setcursortype(_NOCURSOR);
    clrscr();
    int selection = 0,menu=0;
    char ch;
    int first = 1;
    char section[100] = "μ";
}

```

```

if(type!='\0')
{
    strcat(section,type);
    strcat(section," INFORMATION");
    menu=1;
}

char opt[2][4][50] =
{
{"Student Info","Teacher Info","Transport Info","Exit"},
 {"View Record","Add Record","Remove Record","Back"}
};

ch=0;
while(ch!=13)
{
    if(selection!=0 && ch==72)      selection--;
    if(selection!=3 && ch==80)      selection++;

    clrscr();
    int speed=(first==1)?30:0;
    header(speed);
    box(27, 4, 25, 16, speed, 2);
    int ypos=10;

    if(strcmpi(section,"μ")!=0)
    {
        line(2,3,78,24,0,1);
        gotoxy(4,3);      puts(section);
    }

    for(int i=0;i<4;++i)

```

```

{
    gotoxy(35,7);      puts("Choose Option:");
    gotoxy(35,ypos);  puts(opt[menu][i]);
    if(i==selection) box(32,ypos-2,15,2,0);
    ypos+=3;

}

ch=getch();
first =0;
}

if(type=='\0')
{
    switch(selection)
    {
        case 0:    menuHandler("STUDENT");
        break;
        case 1:    menuHandler("TEACHER");
        break;
        case 2:    menuHandler("TRANSPORT");
        break;
        case 3: exit(0);
    }
}
else if(strcmpi(type,"student")==0)
{
    switch(selection)
    {
        case 0:    viewStudent();
        break;
        case 1:    addStudent();
    }
}

```

```

        break;

    case 2:      remRcrd("STUDENT");

        break;

    case 3:

        break;

    }

}

else if(strcmpi(type, "TEACHER")==0)

{

    switch(selection)

    {

        case 0:      viewTeacher();

        break;

        case 1:      addTeacher();

        break;

        case 2:      remRcrd("TEACHER");

        break;

        case 3:

        break;

    }

}

else if(strcmpi(type, "TRANSPORT")==0)

{

    switch(selection)

    {

        case 0:      viewBus();

        break;

        case 1:      addBus();

        break;

        case 2:      remRcrd("TRANSPORT");

        break;

        case 3:

```

```
        break;
    }
}

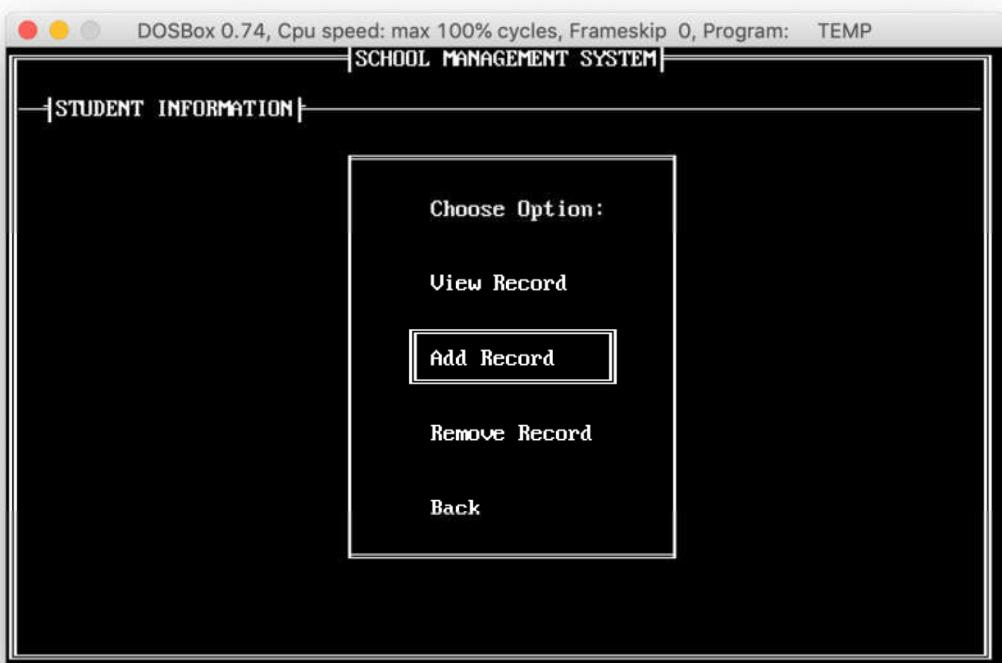
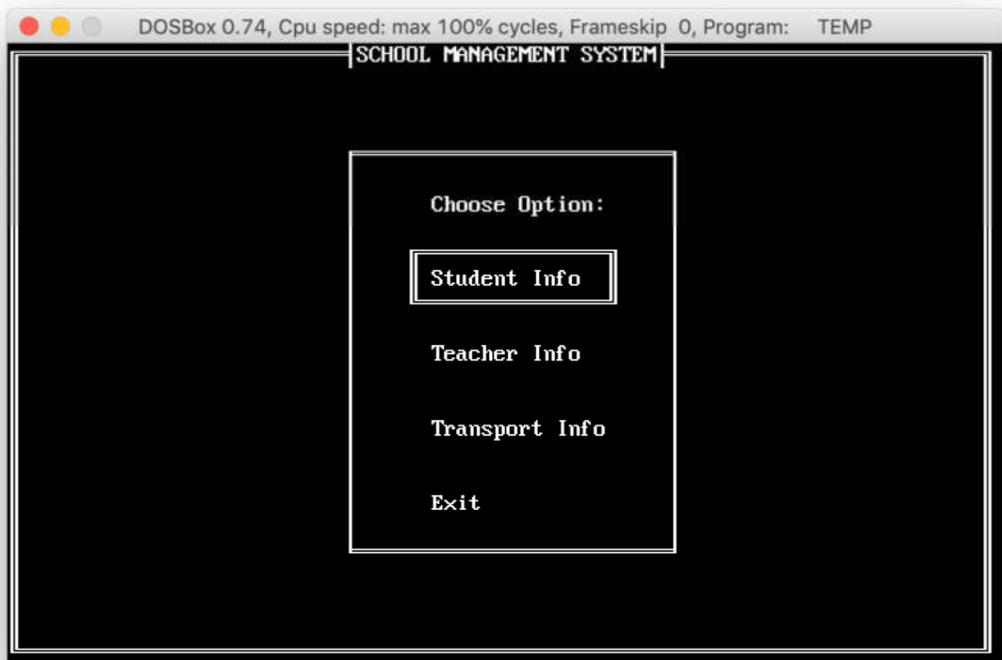
void main()
{
    _setcursortype(_NOCURSOR);
    checkDir();
    constream win;
    win<<setclr(WHITE);
    //addBus();

    //startScreen();           //Enable later
    while(1)
        menuHandler('\0');

}

// End of program
```

OUTPUT



DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TEMP
[SCHOOL MANAGEMENT SYSTEM]

[STUDENT RECORD ADDITION FORM]

Admn No:	Grade:	Section:
Name:	Father's Name:	
Gender (M/F):	Date of Birth: DDMYYYY	
School Transport?(Y/N):		

DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TEMP
[SCHOOL MANAGEMENT SYSTEM]

[STUDENT RECORD ADDITION FORM]

Enter stream:	Subject:	Elected Subject:
<input type="checkbox"/> Science	<input type="checkbox"/> Comp Science	<input type="checkbox"/> Maths
<input type="checkbox"/> Commerce	<input type="checkbox"/> Biology	<input type="checkbox"/> Info

DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TEMP
|SCHOOL MANAGEMENT SYSTEM|

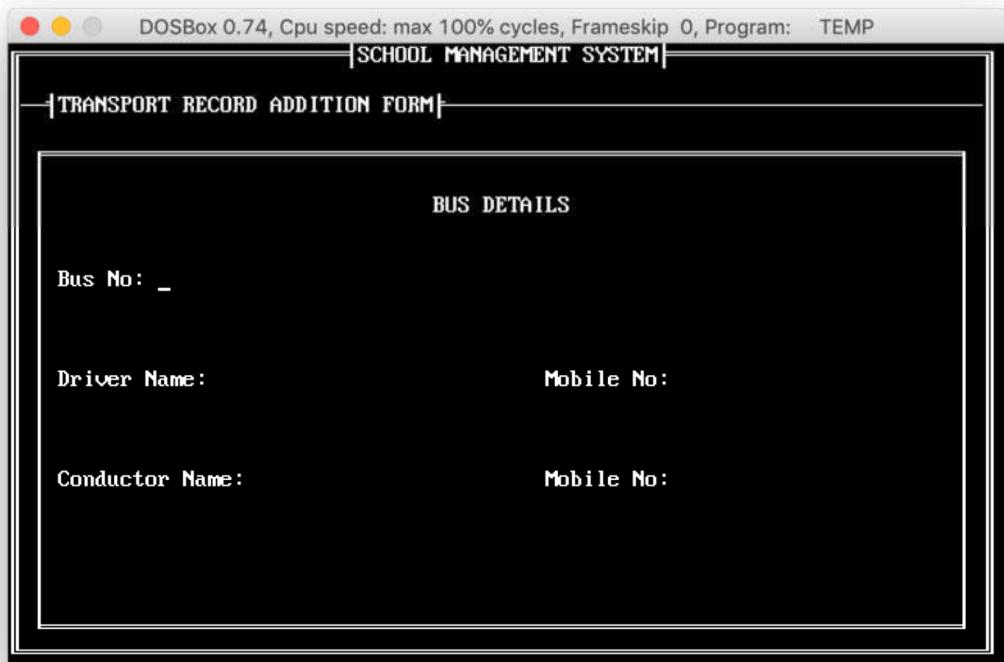
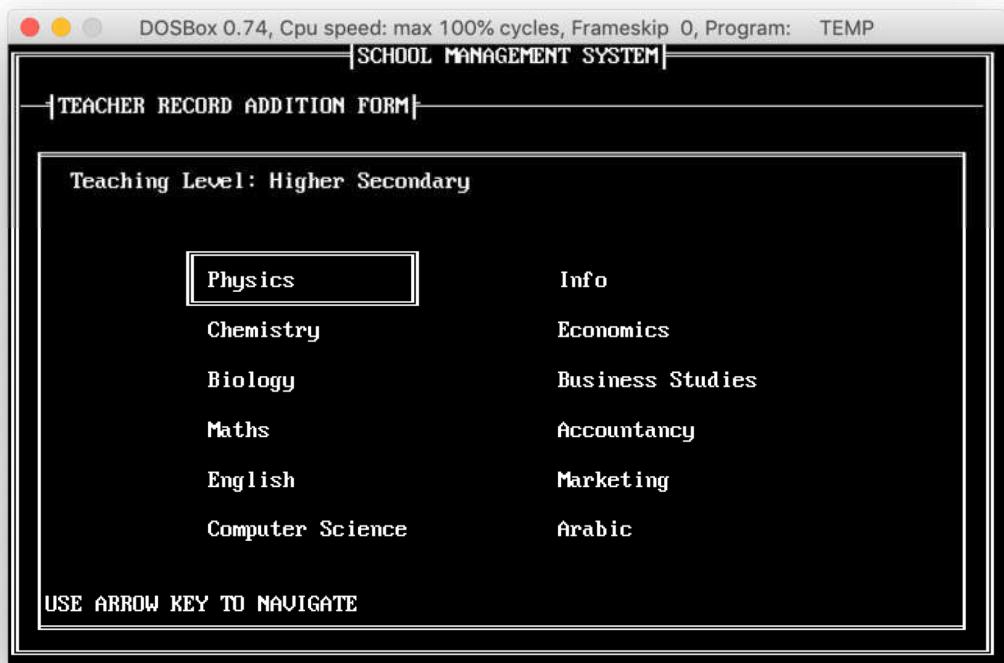
|TEACHER RECORD ADDITION FORM|

Staff ID: _	Mobile No:
First Name:	Last Name:
Date of Birth: DDMMYYYY	Gender (M/F):
School Transport?(Y/N):	Class Teacher?(Y/N):

DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TEMP
|SCHOOL MANAGEMENT SYSTEM|

|TEACHER RECORD ADDITION FORM|

Teaching Level:
Primary
Secondary
Higher Secondary



DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TEMP

SCHOOL MANAGEMENT SYSTEM

|TRANSPORT RECORD ADDITION FORM|

Location:

Sharjah

Dubai

Ajman

Dha'yd

DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TEMP

SCHOOL MANAGEMENT SYSTEM

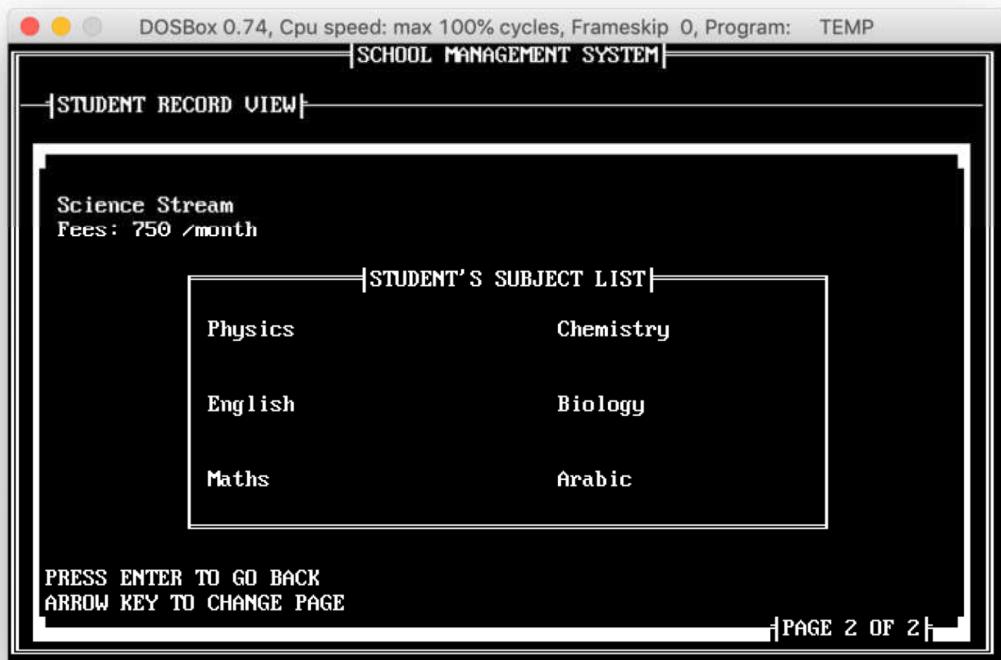
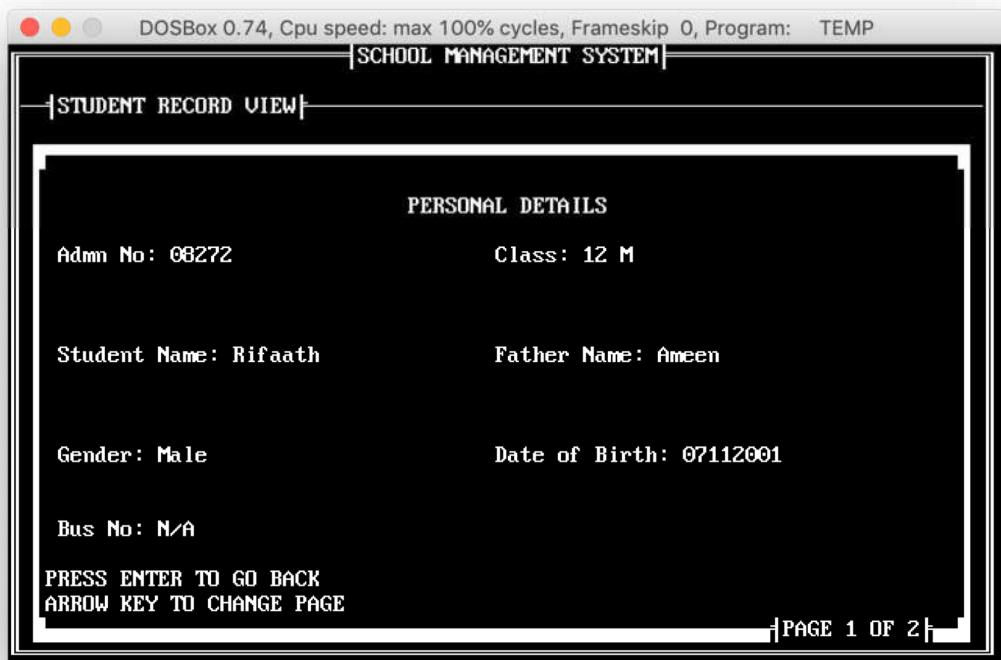
|STUDENT RECORD VIEW FORM|

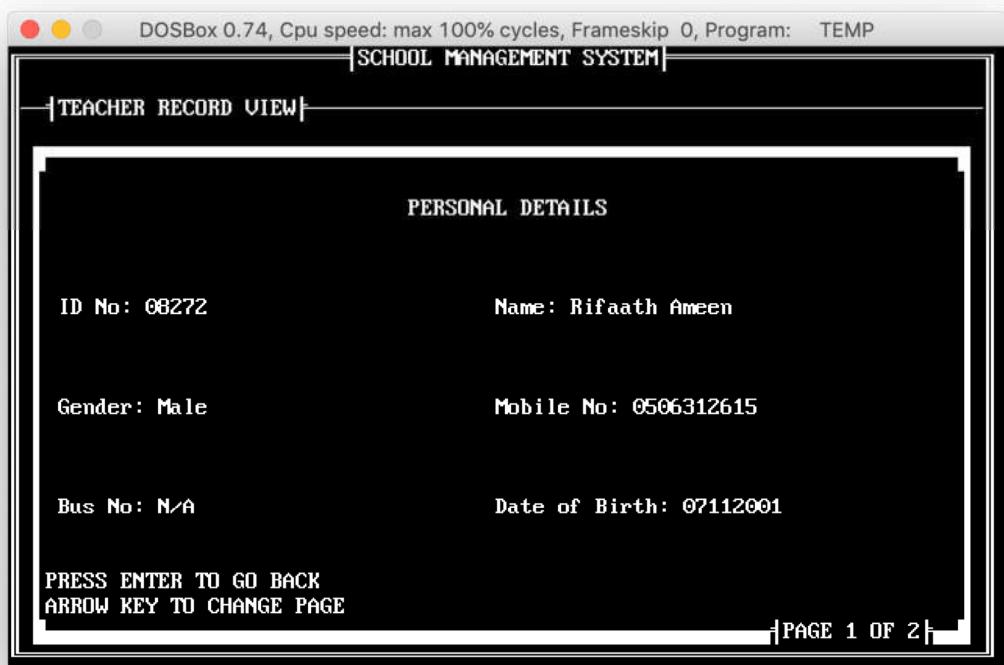
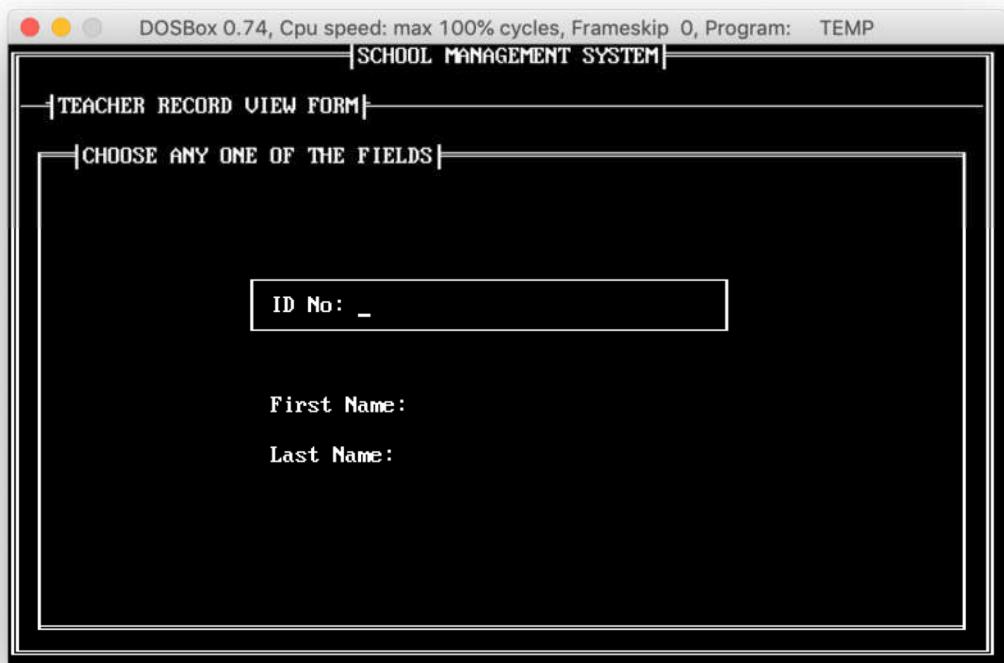
|CHOOSE ANY ONE OF THE FIELDS|

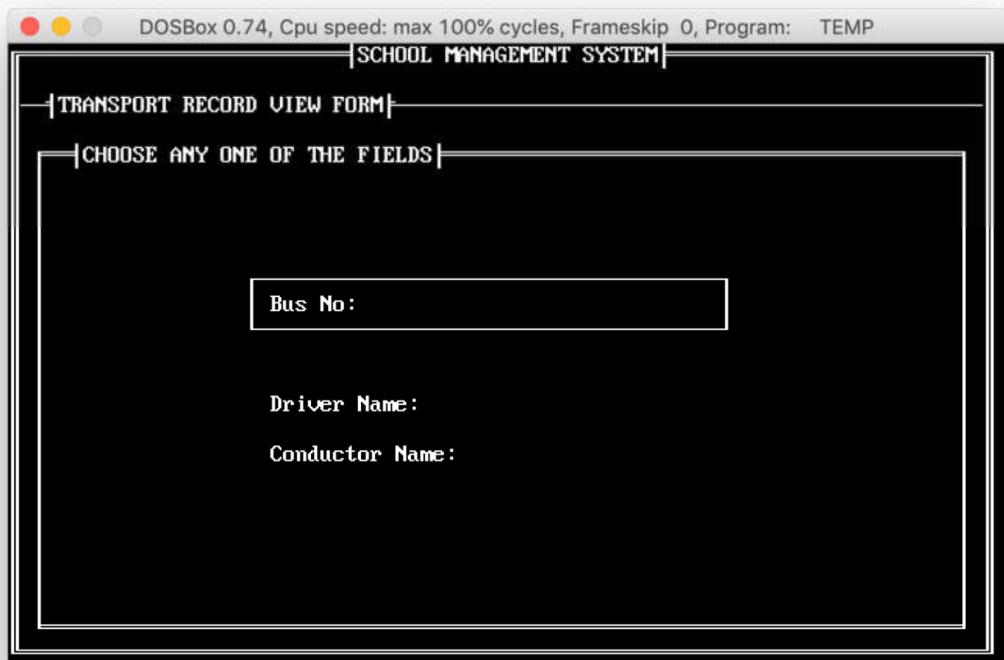
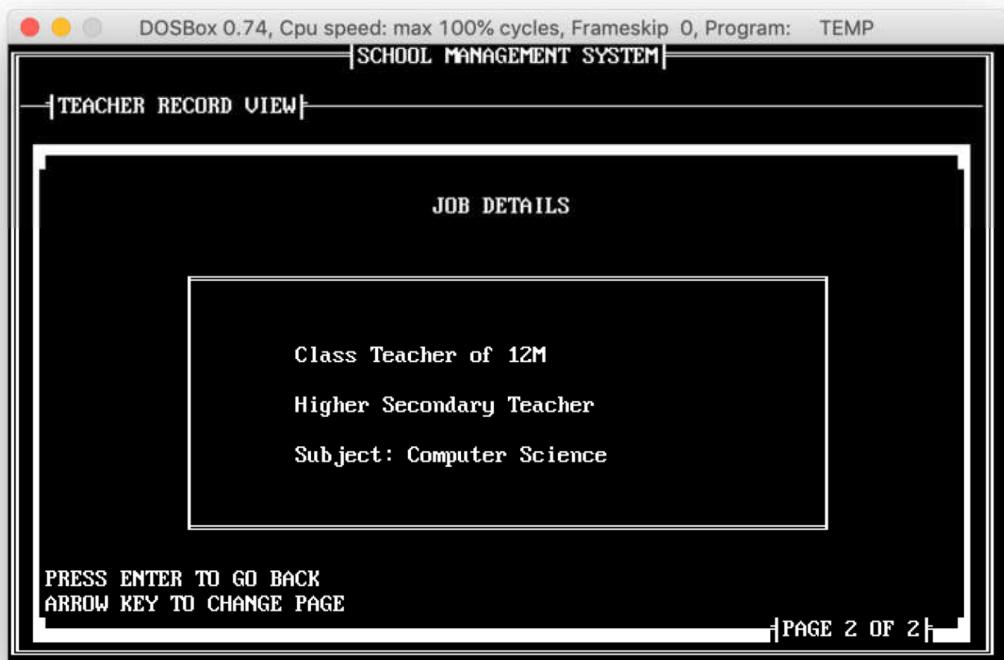
Admn No:

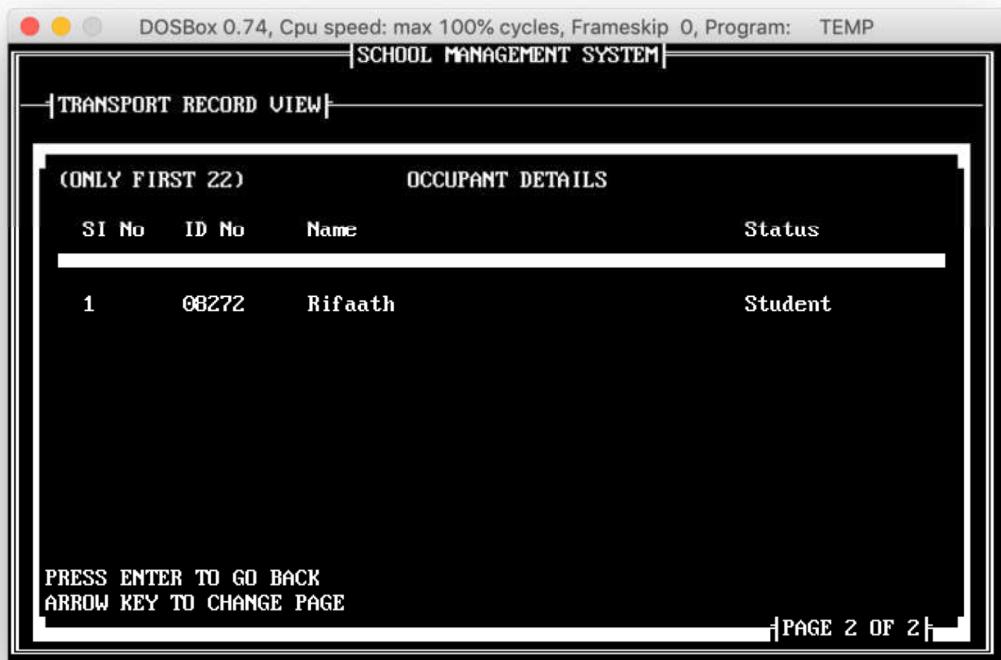
Name:

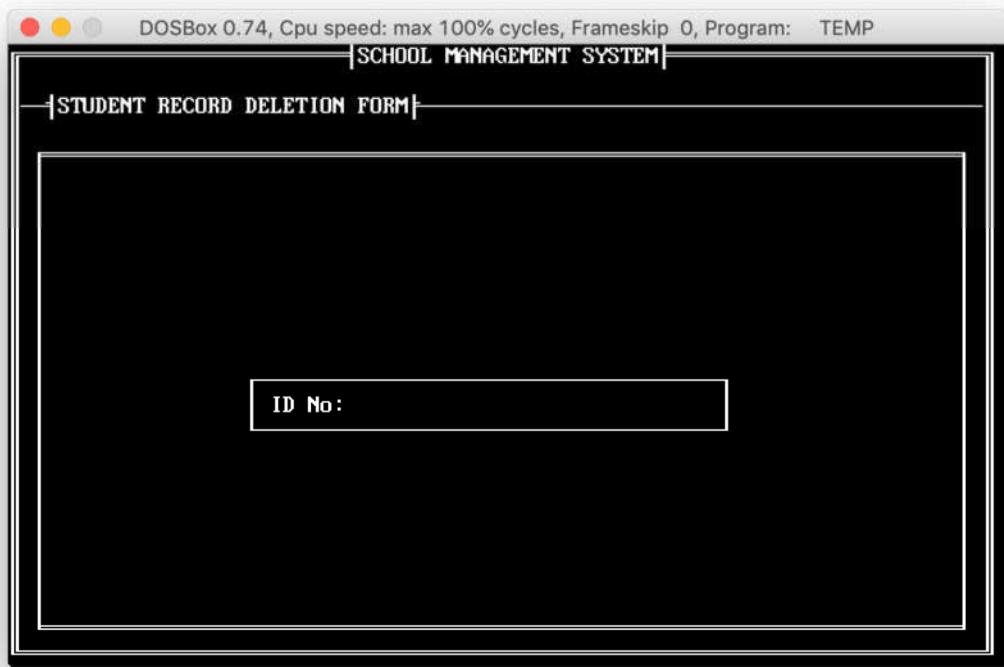
Father's Name:











CONCLUSION

While making the project on "SCHOOL MANAGEMENT" I made my progress by solving several problems. A solution to each problem by myself was the most important part of the project and this provided me with experiences which will help me in the future.

Some important things that I learned include designing a good program architecture and converting real-life situations into an efficient code, and how to write a good looking easily readable and understandable as well as time and memory-efficient code.

At the end of this course, I was able to implement data file handling concepts into real-world systems.

BIBLIOGRAPHY

- www.geeksforgeeks.com
- www.google.com
- [wwwcplusplus.com](http://www.cplusplus.com)