**INDEX**

**S.No. TOPIC PAGE NO.**

1. **INTRODUCTION……………………………………2**
2. **HEADER FILES INCLUDED……………………3-5**
3. **CLASSES USED………………………………....6-7**
4. **SOURCE CODE……………………………………9-39**
5. **OUTPUT SCREEN SNAPSHOTS……………40-44**
6. **ADVANTAGES AND DISADVANTAGES………..45**
7. **BIBLIOGRAPHY……………………………………46**

**INTRODUCTION**

With the advent of computers many tasks of humans have became easier. For example in a car manufacturing industry those high precision metal cutting lasers are controlled by computers for humans have a limited degree of accuracy and efficiency: they cannot cut metal sheets all day long☺ But many of the tasks like designing a new engine or listening to human queries are the work of humans. Hence it be made clear that for the efficient functioning of an organization, the accuracy, precision and efficiency of computers along with the superb intellect of human brain is required, well intermeshed.

Keeping these points in mind, I have developed a program in which a **bus travel agency's records are managed**. This program serves the function of an efficient filing system in which one can **create, modify and delete records/files and even search them.**

Keeping in view the basic features of **object oriented programming**, various features in the program make lots of use of **classes and inheritance**. This not only helped in the construction of those features but also helped in making the source code short (**1342 lines**)

which would otherwise have been very long (**approx. 2500-3500**).

**HEADER FILES INCLUDED**

Header files contain definitions of functions and variables which can be incorporated into any C++ program by using the **pre-processor #**include statement. Standard header files are provided with each compiler, and cover a range of areas, string handling, mathematical, data conversion, printing and reading of variables.

When a header file is stated in the beginning of the program, the compiler becomes prepared to accept and execute functions and variables particular to that header file. This results in simplification of programs and thus helps in getting desired results/outputs in a relatively short time( and with less brainstorming☺).

For example**:** by using the header file**<math.h>** one can easily perform squaring operations or square root operations by use of ***pow()*** and ***sqrt()*** functions defined in the **<math.h>** header file. This makes programming easy and source code short which otherwise would have been tricky and long.

The header files along with their important functions so used in this program are as follows **:**

1. **fstream.h :** This header file is one of the most important header file in C++ programming. Reason being that this header file contains the definitions of functions for file input and output like **'cout'** or **'cin'.**

It is mostly used with **iostream.h** header file which is inherited from **fstream.h** only and basically contains functions for simple I/O operations.

Some important functions and variables and macros used in this program which are defined by fstream.h and iostream.h are as follows **:**

1. **ifstream()**  and **istream():**  for file input(or say file to program) operations. Some associated file modes to these functions are **ios::in,ios::ate, ios::binary.**
2. **ofstream()**  and  **ostream():** for file output(or say program to file) operations. Some file modes associated to it are **ios::out,ios::trunc,ios::nocreate,ios::noreplace,ios::app,ios::ate** and **ios::binary.**
3. **fstream():** for both file input and output operations. It can use all the file modes associated to the input and output operations in file.
4. **cout:** for display of a statement on screen
5. **cin:** for input of values in a variable
6. **stdio.h :** This header is also an important one as it primarily provides various functions for input and output of string. For example :
7. **gets()** and **get():** These functions are helpful for input of a string of charcters(like words) and a single character into the program respectively.
8. **puts()**  and **put():** These functions are used for output of a string of characters and a single character from the program respectively.
9. **getline():** This function is used for input of a string of words( like sentences ) into the program. This generally helps in reading an entire line from a text file.

1. **conio.h:** This function is generally used for controlling the display of outputs on a screen. Its important functions used in the program are described below :
2. **clrscr():** This function serves the purpose of clearing up the contencts of an entire screen i.e., having a blank screen.
3. **getch():** This functions stabilizes the output on the screen by making the system wait for the user to enter a key to proceed to next output screen.
4. **gotoxy():** This function is used for random cursor and output placement on screen. Its genral form is gotoxy(x,y), where x is the x(horizontal) coordinate of the screen and y is the y(vertical) coordinate of the screen. Note that the (0,0) coordinate fo the screen lies at the topmost left corner of the screen.

gotoxy(0,0)

1. **string.h:** This header files is used for manipulation of strings. Two of the most important of its functions used in the program are:
2. **strcpy(s1,s2):** This function copies value of s2 to s1.
3. **strcmp(s1,s2):** This function compares the value of s1 and s2. It returns 0 if both strings are the same. Otherwise ib condition of a mismatch it returns a non zero value.
4. **process.h:**  was used for the exit(0) function, which makes the user exit the program.
5. **dos.h:** is used to use delay() function which delays the next operation in the program by the time entered in it in milliseconds.

For ex: delay(3000);

will delay the next operation in program by 3 seconds.

**Classes used**

Class is a way to bind the data describing an entity and its associated functions together. In C++, class makes a data type which is used to create objects of this type. By doing so the program follows the concept of data abstraction. Furthermore classes can be inherited i.e., properties of one class(s) can be taken up by other class(s). This allows for code reusability and saving of time and not to soy of efficiency in managing the program.

In my program too, lots of classes integrated with each other by virtue of inheritance has been used each one of them I am describing in brief over here :

1. **basicfunctions:**  This class contains functions which can perform basic operations on several text files. These operations are:
2. **opening a file** which is done by **int opentxtf(char filen[],int size);**
3. **modifying a file**  which contains two options : modifying a line in afile and adding aline in the file. These two operations are done by **int modifymenu();**, **int modifytxtf();** and **int addlinetxtf();.**
4. **deleting a file** which is done by **void deletetxtf();.**
5. 4 classes inherit the class **basicfunctions.** These classes create and manage text files. These classes are **driver, conductor, other\_employee** and **bus**  and they manage text files related to driver , conductor , other employees and bus respectively.
6. **customer\_rec** class manages the record of customer and is binary file. As binary files are different in processing by computer than text files, hence this class is solely separated files from all other classes. Binary file operations in this class are managed by 4 different functions, namely **void customer\_enter();**,

**void customer\_view();, void customer\_modify();** and

**void customer\_delete();.**

Hence the total number of classes used in this program are **6.**

It would be worthwhile to mention the data files management in the given program. In this program, 4 text files under the category of **driver, conductor, other employee** and **bus**  can be cretated. However each record is different and stored as ad ifferent name so as to make the management of a single element of the organistaion easier. But in the **customer\_rec** class system, only one file,that too binary , with the name **record.dat** is managed.

BUS

**BASICFUNCTIONS**

OTHER\_EMPLOYEE

CUSTOMER\_REC

CONDUCTOR

DRIVER

**THE SOURCE CODE**

#include<fstream.h>

#include<iostream.h>

#include<conio.h>

#include<stdio.h>

#include<string.h>

#include<process.h>

#include<dos.h>

void securityfunc();

void menu();

void loadstart();

void loaderfunc();

void customer\_enter();

void customer\_view();

void customer\_modify();

void customer\_delete();

void securityfunc()

{ char mcode[]={'a','i','b','u','s','1','0','b','#','\0'};

char code[10];

int try=0;

gotoxy(23,24);

cout<<"FOR SECURITY REASONS YOU ARE REQUIRED TO";

tryagain:

gotoxy(23,26);

cout<<"ENTER THE PASSWORD TO ACCESS THE PROGRAM";

lasttry:

gotoxy(37,30);

cout<<"\_\_\_\_\_\_\_\_\_";

gotoxy(37,29);

gets(code);

++try;

if(strcmp(code,mcode)==0)

{ clrscr();

gotoxy(38,26);

cout<<"CORRECT!!";

delay(1500);

loadstart();

loaderfunc();

menu();

}

else if(try<3)

{ clrscr();

gotoxy(38,26);

cout<<"WRONG!!";

gotoxy(28,28);

cout<<"Do you wish to try again?(y/n)";

char ans;

gotoxy(20,30);

cout<<"-> ";

cin>>ans;

if(ans=='y'||ans=='Y')

{ clrscr();

goto tryagain;

}

else

{ clrscr();

gotoxy(25,28);

cout<<"YOU ARE NOW EXITING OUT OF PROGRAM";

delay(3000);

exit(0);

}

}

else if(try==3)

{ clrscr();

gotoxy(20,24);

cout<<"You have tried 3 times to enter into this program."

<<"\t\t\t\tThis is your last chance.Be careful!!\n";

goto lasttry;

}

else

{ clrscr();

gotoxy(25,25);

cout<<"YOU ARE NOW EXITING OUT OF PROGRAM";

delay(3000);

exit(0);

}

}

void loadstart()

{ clrscr();

for(int i=0;i<80;i++)

{ clrscr();

gotoxy(38,22);

cout<<"LOADING";

gotoxy(i,24);

cout<<"---------";

gotoxy(i,25);

cout<<"|\_:\_:\_:\_\_|";

gotoxy(i,26);

cout<<"| |";

gotoxy(i,26);

cout<<"\_\_\_\_\_\_\_\_\_";

gotoxy(i,27);

cout<<" o o ";

delay(50);

}

}

void loaderfunc()

{ clrscr();

for(int i=0;i<40;i++)

{ clrscr();

gotoxy(i,20);

cout<<"WELCOME";

delay(100);

if(i==39)

{ for(int j=75;j>41;j--)

{ clrscr();

gotoxy(39,20);

cout<<"WELCOME";

gotoxy(j,23);

cout<<"TO";

delay(100);

}

}

}

gotoxy(38,26);

cout<<"TRAVEL X";

delay(1000);

clrscr();

gotoxy(38,20);

cout<<"TRAVEL X";

gotoxy(38,21);

cout<<"\_\_\_\_\_\_\_\_";

gotoxy(26,23);

cout<<"Your Travel Systems Manager Program";

gotoxy(26,24);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(36,28);

cout<<"<BUS VERSION>";

delay(2000);

}

class basicfunctions

{ char filename[20],choice[4],choice1,lineinput[70],pgcommand[4],newline[70];

int lineno,linenoI,newlineno,chk;

public:

basicfunctions()

{ lineno=0;

newlineno=0;

};

int opentxtf(char filen[],int size);

int modifymenu();

int modifytxtf();

int addlinetxtf();

void deletetxtf();

};

int basicfunctions::opentxtf(char filen[],int size)

{ clrscr();

if(filen[size]!=0)

{ strcpy(filename,filen);

goto ex;

}

lineno=1;

cout<<"\n Enter the name of the you wish to open\n(use'.txt'extension): ";

gets(filename);

ex:

ifstream opentxtf(filename,ios::in|ios::app|ios::nocreate);

if(!opentxtf)

{ clrscr();

gotoxy(32,24);

cout<<"FILE DOESN'T";

gotoxy(36,26);

cout<<"EXISTS!!";

delay(3000);

clrscr();

gotoxy(1,23);

cout<<"Do you wish to create one?(y/n)\n ->";

gets(choice);

if(strcmp(choice,"y")==0||strcmp(choice,"Y")==0)

return -1;

else

return 0;

}

else

{ clrscr();

cout<<filename

<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

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

while(!opentxtf.eof())

{ ++lineno;

opentxtf.getline(lineinput,68);

cout<<lineno<<". "<<lineinput<<endl;

}

lineno=0;

cout<<"Press any key to go back";

getch();

opentxtf.close();

return 0;

}

}

int basicfunctions::modifymenu()

{ clrscr();

cout<<"<-";

gotoxy(37,20);

cout<<"MODIFY MENU";

gotoxy(27,21);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(28,23);

cout<<"1. Replace and change a line";

gotoxy(28,25);

cout<<"2. Add a new line";

gotoxy(27,26);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

cout<<"\n\n Enter Command : ";

gets(choice);

if(strcmp(choice,"<-")==0)

return 0;

else if(strcmp(choice,"1")==0)

{ int r1=modifytxtf();

return r1;

}

else if(strcmp(choice,"2")==0)

{ int r2=addlinetxtf();

return r2;

}

return 0;

}

int basicfunctions::modifytxtf()

{ MF:

clrscr();

cout<<"<- ->\n"

<<"\n Enter file name\n(Use'.txt' extension) : ";

gets(filename);

fstream apptxtf(filename,ios::in|ios::out|ios::app);

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

ofstream newtxtf("tcc.txt",ios::out);

if(!apptxtf)

{ clrscr();

gotoxy(24,24);

cout<<"FILE DOESN'T";

gotoxy(26,26);

cout<<"EXISTS !!";

delay(3000);

clrscr();

gotoxy(1,23);

cout<<"Do you wish to create one?(y/n)\n ->";

cin>>choice1;

if(choice1=='y'||choice1=='Y')

return -1;

else

return 0;

}

else

{ cout<<filename<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" It is recommended that you first note down the line number"

<<"\n Do you wish to view "<<filename<<" for that purpose first?(y/n)";

cin>>choice1;

if(choice1=='y'||choice1=='Y')

{ clrscr();

gotoxy(10,24);

cout<<" File "<<filename<<" is being opened";

gotoxy(10,26);

cout<<"Press any key to continue";

clrscr();

cout<<"\n\n See the line number to be changed";

delay(2000);

opentxtf(filename,20);

lineno=1;

}

else

gotoxy(32,24);

cout<<"OKAY";

}

ML:

clrscr();

cout<<"<- \nModifying "<<filename<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<"Enter the lineno. of the line you wish you wish to add : ";

cin>>newlineno;

while(!apptxtf.eof())

{ apptxtf.getline(lineinput,70);

if(lineno==newlineno)

{ cout<<"Enter the line you wish to add : \n";

gets(newline);

newtxtf<<newline<<endl;

}

else

newtxtf<<lineinput<<endl;

lineno++;

}

remove(filename);

rename("tcc.txt",filename);

apptxtf.close();

newtxtf.close();

lineno=0;

clrscr();

gotoxy(20,25);

cout<<"Press 1 to add another line";

gotoxy(20,27);

cout<<"Press 2 to add line to a different file";

gotoxy(20,29);

cout<<"Enter '<-' to go back to the respective menu";

gotoxy(20,31);

cout<<" Enter Command : ";

gets(choice);

if(strcmp(choice,"<-")==0)

return 0;

else if(strcmp(choice,"1")==0)

goto ML;

else if(strcmp(choice,"2")==0)

goto MF;

return 0;

}

void basicfunctions::deletetxtf()

{ MF:

clrscr();

cout<<" \n\n Enter the name of the file you wish to delete(use .txt extension) : ";

gets(filename);

ifstream deletex(filename,ios::in);

while(!deletex.eof())

{ deletex.getline(lineinput,70);

++lineno;

cout<<"\n"<<lineno<<". "<<lineinput<<"\n";

}

cout<<"\nAre you sure that you wish to delete the file?(y/n)";

gets(choice);

if(strcmp(choice,"y")==0||strcmp(choice,"Y"))

remove(filename);

else

{ clrscr();

gotoxy(26,24);

cout<<"\a GOING BACK ";

return ;

}

}

int basicfunctions::addlinetxtf()

{ MF:

clrscr();

cout<<"<- ->\n"

<<"\n Enter file name\n(Use'.txt' extension) : ";

gets(filename);

fstream apptxtf(filename,ios::in|ios::out|ios::app);

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

ofstream newtxtf("tcc.txt",ios::out);

if(!apptxtf)

{ clrscr();

gotoxy(24,24);

cout<<"FILE DOESN'T";

gotoxy(26,26);

cout<<"EXISTS !!";

delay(3000);

clrscr();

gotoxy(1,23);

cout<<"Do you wish to create one?(y/n)\n ->";

cin>>choice1;

if(choice1=='y'||choice1=='Y')

return -1;

else

return 0;

}

else

{ cout<<filename<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<" It is recommended that you first note down the line number"

<<"\n Do you wish to view "<<filename<<" for that purpose first?(y/n)";

cin>>choice1;

if(choice1=='y'||choice1=='Y')

{ clrscr();

gotoxy(10,24);

cout<<" File "<<filename<<" is being opened";

gotoxy(10,26);

cout<<"Press any key to continue";

clrscr();

cout<<"\n\n See the line number to be changed";

delay(2000);

opentxtf(filename,20);

lineno=1;

}

else

gotoxy(32,24);

cout<<"OKAY";

}

ML:

clrscr();

cout<<"<- \nModifying "<<filename<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

cout<<"Enter the lineno. of the line you wish you wish to add : ";

cin>>newlineno;

while(!apptxtf.eof())

{ apptxtf.getline(lineinput,70);

if(lineno==newlineno)

{ cout<<"Enter the line you wish to add : \n";

gets(newline);

newtxtf<<newline<<endl;

newtxtf<<lineinput<<endl;

}

else

newtxtf<<lineinput<<endl;

lineno++;

}

remove(filename);

rename("tcc.txt",filename);

apptxtf.close();

newtxtf.close();

lineno=0;

clrscr();

gotoxy(20,25);

cout<<"Press 1 to add another line";

gotoxy(20,27);

cout<<"Press 2 to add line to a different file";

gotoxy(20,29);

cout<<"Enter '<-' to go back to the respective menu";

gotoxy(20,31);

cout<<" Enter Command : ";

gets(choice);

if(strcmp(choice,"<-")==0)

return 0;

else if(strcmp(choice,"1")==0)

goto ML;

else if(strcmp(choice,"2")==0)

goto MF;

return 0;

}

//CLASS BASICFUNCTIONS' DEFINITIONS END HERE

class driver:public basicfunctions

{ //Variable set

//Detail Set 1

char filename1[20],name[40],mname[40],fname[40],dob[12],doj[12],bgroup[3];

//Detail Set 2

char residence[60],licenseno[20];

long int phone\_no;

long int aadharno;

//Details dealing with skills and criminal record

char education[60],permit[60],crime\_rec[70];

//Salary

float salary;

//Other details

char other\_det[70];

char Dchoice[4],choice;

public:

int driver\_menu();

int driver\_createf();

};

int driver::driver\_menu()

{ clrscr();

Dloop:

clrscr();

cout<<"<-";

gotoxy(32,12);

cout<<"DRIVER MENU";

gotoxy(31,14);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(32,16);

cout<<"1. Create File";

gotoxy(32,18);

cout<<"2. View File";

gotoxy(32,20);

cout<<"3. Modify File";

gotoxy(32,22);

cout<<"4. Delete File";

gotoxy(31,24);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(31,26);

cout<<"Enter Command : ";

gets(Dchoice);

if(strcmp(Dchoice,"<-")==0)

return 0;

else if(strcmp(Dchoice,"1")==0)

{ driver\_createf();

goto Dloop;

}

else if(strcmp(Dchoice,"2")==0)

{ int rod=opentxtf(0,0);

if(rod==0)

goto Dloop;

else

driver\_createf();

}

else if(strcmp(Dchoice,"3")==0)

{ int mod=modifymenu();

if(mod==0)

goto Dloop;

else

driver\_createf();

}

else if(strcmp(Dchoice,"4")==0)

{ deletetxtf();

goto Dloop;

}

return 0;

}

int driver::driver\_createf()

{ rpp:

clrscr();

cout<<"New File\n\n\n Enter the name the with which you wish to save the file(use '.txt' extension) : ";

gets(filename1);

clrscr();

cout<<filename1<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

ofstream createstrm(filename1,ios::out);

cout<<"\nEnter the name of the driver : ";

gets(name);

createstrm<<" Name : "<<name;

cout<<"\nEnter the name of Mother of the driver : ";

gets(mname);

createstrm<<"\n Mother's name : "<<mname;

cout<<"\nEnter the name of Father of the driver : ";

gets(fname);

createstrm<<"\n Father's name : "<<fname;

cout<<"\nEnter date of birth of the driver : ";

gets(dob);

createstrm<<"\n Date of Birth : "<<dob;

cout<<"\nEnter blood group of the driver : ";

gets(bgroup);

createstrm<<"\n Blood Group : "<<bgroup;

cout<<"\nEnter residence address of the driver : ";

gets(residence);

createstrm<<"\n Residence Address : "<<residence;

cout<<"\nEnter Aadhar No.of the driver :";

cin>>aadharno;

createstrm<<"\n Aadhar Number : "<<aadharno;

cout<<"\nEnter License No. of the driver : ";

gets(licenseno);

createstrm<<"\n License Number : "<<licenseno;

cout<<"\n Phone Number : ";

cin>>phone\_no;

createstrm<<"\n Phone Number : "<<phone\_no;

cout<<"\nEnter the maximum academic qualification of the driver : ";

gets(education);

createstrm<<"\n Maximum academic qualification : "<<education;

cout<<"\nEnter the date of joining of the driver : ";

gets(doj);

createstrm<<"\n Date of Joining : "<<doj;

cout<<"\nEnter the permit of the driver : ";

gets(permit);

createstrm<<"\n Permit : "<<permit;

cout<<"\n Enter criminal record,if any,to which the driver was a part : ";

gets(crime\_rec);

createstrm<<"\n Criminal Record : "<<crime\_rec;

cout<<"\nOther details\n";

createstrm<<"\nOther details";

choice='y';

while(choice=='y'||choice=='Y')

{ cout<<": ";

gets(other\_det);

createstrm<<"\n"<<other\_det;

cout<<"\n\n Do you wish to enter another line?(y/n)";

cin>>choice;

}

clrscr();

createstrm.close();

cout<<"Do you wish to create another file?(Y/N)";

cin>>choice;

if(choice=='y'||choice=='Y')

goto rpp;

else

return 0;

}

class conductor:public basicfunctions

{ //Variable set

//Detail Set 1

char filename1[20],name[40],mname[40],fname[40],dob[12],doj[12],bgroup[3];

//Detail Set 2

char residence[60],licenseno[20];

long int phone\_no;

long int aadharno;

//Details dealing with skills and criminal record

char education[60],permit[60],crime\_rec[70];

//Salary

float salary;

//Other details

char other\_det[70];

char Dchoice[4],choice;

public:

int conductor\_menu();

int conductor\_createf();

};

int conductor::conductor\_menu()

{ clrscr();

Dloop:

clrscr();

cout<<"<-";

gotoxy(32,12);

cout<<"CONDUCTOR MENU";

gotoxy(31,14);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(32,16);

cout<<"1. Create File";

gotoxy(32,18);

cout<<"2. View File";

gotoxy(32,20);

cout<<"3. Modify File";

gotoxy(32,22);

cout<<"4. Delete File";

gotoxy(31,24);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(31,26);

cout<<"Enter Command : ";

gets(Dchoice);

if(strcmp(Dchoice,"<-")==0)

return 0;

else if(strcmp(Dchoice,"1")==0)

{ conductor\_createf();

goto Dloop;

}

else if(strcmp(Dchoice,"2")==0)

{ int rod=opentxtf(0,0);

if(rod==0)

goto Dloop;

else

conductor\_createf();

}

else if(strcmp(Dchoice,"3")==0)

{ int mod=modifymenu();

if(mod==0)

goto Dloop;

else

conductor\_createf();

}

else if(strcmp(Dchoice,"4")==0)

{ deletetxtf();

goto Dloop;

}

return 0;

}

int conductor::conductor\_createf()

{ rpp:

clrscr();

cout<<"New File\n\n\n Enter the name the with which you wish to save the file(use '.txt' extension) : ";

gets(filename1);

clrscr();

cout<<filename1<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

ofstream createstrm(filename1,ios::out|ios::app);

cout<<"\nEnter the name of the conductor : ";

gets(name);

createstrm<<" Name : "<<name;

cout<<"\nEnter the name of Mother of the conductor : ";

gets(mname);

createstrm<<"\n Mother's Name : "<<mname;

cout<<"\nEnter the name of Father of the conductor : ";

gets(fname);

createstrm<<"\n Father's Name : "<<fname;

cout<<"\nEnter date of birth of the conductor : ";

gets(dob);

createstrm<<"\n Date of Birth : "<<dob;

cout<<"\nEnter blood group of the conductor : ";

gets(bgroup);

createstrm<<"\n Blood Group : "<<bgroup;

cout<<"\nEnter residence address of the conductor : ";

gets(residence);

createstrm<<"\n Residence Address : "<<residence;

cout<<"\nEnter Aadhar No.of the conductor :";

cin>>aadharno;

createstrm<<"\n Aadhar Number : "<<aadharno;

cout<<"\nEnter License No. of the conductor : ";

gets(licenseno);

createstrm<<"\n License Number : "<<licenseno;

cout<<"\n Phone Number : ";

cin>>phone\_no;

createstrm<<"\nPhone Number : "<<phone\_no;

cout<<"\nEnter the maximum academic qualification of the conductor : ";

gets(education);

createstrm<<"\n Maximum academic qualification : "<<education;

cout<<"\nEnter the date of joining of the conductor : ";

gets(doj);

createstrm<<"\n Date of Joining : "<<doj;

cout<<"\nEnter the permit of the conductor : ";

gets(permit);

createstrm<<"\n Permit : "<<permit;

cout<<"\n Enter criminal record,if any,to which the conductor was a part : ";

gets(crime\_rec);

createstrm<<"\n Criminal Record : "<<crime\_rec;

cout<<"\nOther details\n";

createstrm<<"\n Other Details ";

choice='y';

while(choice=='y'||choice=='Y')

{ cout<<": ";

gets(other\_det);

createstrm<<"\n"<<other\_det;

cout<<"\n\n Do you wish to enter another line?(y/n)";

cin>>choice;

}

clrscr();

createstrm.close();

cout<<"Do you wish to create another file?(Y/N)";

cin>>choice;

if(choice=='y'||choice=='Y')

goto rpp;

else

return 0;

}

class other\_employee:public basicfunctions

{ //Variable set

//Detail Set 1

char filename1[20],name[40],mname[40],fname[40],dob[12],doj[12],bgroup[3],occ[15];

//Detail Set 2

char residence[60],licenseno[25];

long int phone\_no;

long int aadharno;

//Details dealing with skills and criminal record

char education[60],crime\_rec[70];

//Salary

float salary;

//Other details

char other\_det[70];

char Dchoice[4],choice;

public:

int otemp\_menu();

int otemp\_createf();

};

int other\_employee::otemp\_menu()

{ clrscr();

Dloop:

clrscr();

cout<<"<-";

gotoxy(33,12);

cout<<"OTHER EMPLOYEES MENU";

gotoxy(31,14);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(32,16);

cout<<"1. Create File";

gotoxy(32,18);

cout<<"2. View File";

gotoxy(32,20);

cout<<"3. Modify File";

gotoxy(32,22);

cout<<"4. Delete File";

gotoxy(31,24);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(31,26);

cout<<"Enter Command : ";

gets(Dchoice);

if(strcmp(Dchoice,"<-")==0)

return 0;

else if(strcmp(Dchoice,"1")==0)

{ otemp\_createf();

goto Dloop;

}

else if(strcmp(Dchoice,"2")==0)

{ int rod=opentxtf(0,0);

if(rod==0)

goto Dloop;

else

otemp\_createf();

}

else if(strcmp(Dchoice,"3")==0)

{ int mod=modifymenu();

if(mod==0)

goto Dloop;

else

otemp\_createf();

}

else if(strcmp(Dchoice,"4")==0)

{ deletetxtf();

goto Dloop;

}

return 0;

}

int other\_employee::otemp\_createf()

{ rpp:

clrscr();

cout<<"New File\n\n\n Enter the name the with which you wish to save the file(use '.txt' extension) : ";

gets(filename1);

clrscr();

cout<<filename1<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

ofstream createstrm(filename1,ios::out|ios::app);

cout<<"\nEnter the name of the employee : ";

gets(name);

createstrm<<" Name : "<<name;

cout<<"\nEnter the name of Mother of the employee : ";

gets(mname);

createstrm<<"\n Mother's Name : "<<mname;

cout<<"\nEnter the name of Father of the employee : ";

gets(fname);

createstrm<<"\n Father's Name : "<<fname;

cout<<"\nEnter date of birth of the employee : ";

gets(dob);

createstrm<<"\n Date of Birth : "<<dob;

cout<<"\nEnter blood group of the employee : ";

gets(bgroup);

createstrm<<"\n Blood Group : "<<bgroup;

cout<<"\nEnter residence address of the employee : ";

gets(residence);

createstrm<<"\n Residnce Address :"<<residence;

cout<<"\nEnter Aadhar No.of the employee :";

cin>>aadharno;

createstrm<<"\n\n Aadhar Number : "<<aadharno;

cout<<"\nEnter License No. of the employee : ";

gets(licenseno);

createstrm<<"\n License Number : "<<licenseno;

cout<<"\n Phone Number : ";

cin>>phone\_no;

createstrm<<"\n Phone Number : "<<phone\_no;

cout<<"\nEnter the maximum academic qualification of the employee : ";

gets(education);

createstrm<<"\n Maximum academic qualification : "<<education;

cout<<"\nEnter the date of joining of the employee : ";

gets(doj);

createstrm<<"\n Date of Joining : "<<doj;

cout<<"\nEnter the job done by employee : ";

gets(occ);

createstrm<<"\n Occupation : "<<occ;

cout<<"\n Enter criminal record,if any,to which the employee was a part : ";

gets(crime\_rec);

createstrm<<"\n Criminal Record : "<<crime\_rec;

cout<<"\nOther details\n";

createstrm<<"\n Other Details ";

choice='y';

while(choice=='y'||choice=='Y')

{ cout<<": ";

gets(other\_det);

createstrm<<"\n "<<other\_det;

cout<<"\n\n Do you wish to enter another line?(y/n)";

cin>>choice;

}

createstrm.close();

clrscr();

cout<<"Do you wish to create another file?(Y/N)";

cin>>choice;

if(choice=='y'||choice=='Y')

goto rpp;

else

return 0;

}

class bus:public basicfunctions

{ int busno,pascap;

char type[10],choice,filename1[20],manufactured\_by[15],manufactured\_on[10],bo[10], registration\_no[15];

char permit[60],contractno[15],condition[70],otherbdet[70],Dchoice[4];

float cost,fucap;

public:

int bus\_createf();

int bus\_menu();

};

int bus::bus\_createf()

{ rpp:

clrscr();

cout<<"New File\n\n\n Enter the name the with which you wish to save the file(use '.txt' extension) : ";

gets(filename1);

clrscr();

cout<<filename1<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

ofstream createstrm(filename1,ios::out|ios::app);

cout<<"\n\n Enter Bus number : ";

cin>>busno;

createstrm<<"\n Bus Number : "<<busno;

cout<<"\nEnter the type of bus : ";

gets(type);

createstrm<<"\n Bus type : "<<type;

cout<<"\n\nPassenge capacity of bus : ";

cin>>pascap;

createstrm<<"\n Passenger Capacity : "<<pascap;

cout<<"\n\n Enter the fuel capacity of bus( in Litres) :";

cin>>fucap;

createstrm<<"\n Fuel capacity( in Litres) : "<<fucap;

cout<<" \n\nManufatured by : ";

gets(manufactured\_by);

createstrm<<"\n Manufactured by : "<<manufactured\_by;

cout<<"Manufacture completed on : ";

gets(manufactured\_on);

createstrm<<"\n Manufacture completed on : "<<manufactured\_on;

cout<<"\n\n Bought on : ";

gets(bo);

createstrm<<"\n Bought on : "<<bo;

cout<<"\n\n Cost : Rs ";

cin>>cost;

createstrm<<"\n Cost : Rs"<<cost;

cout<<" \n\n Contract number : ";

gets(contractno);

createstrm<<"\n Contract number : "<<contractno;

cout<<"\n\n Permit : ";

gets(permit);

createstrm<<"\n Permit : "<<permit;

cout<<"\n\n Condition : ";

gets(condition);

createstrm<<"\n Condition : "<<condition;

clrscr();

createstrm.close();

cout<<"Do you wish to create another file?(Y/N)";

cin>>choice;

if(choice=='y'||choice=='Y')

goto rpp;

else

return 0;

}

int bus::bus\_menu()

{ clrscr();

Dloop:

clrscr();

cout<<"<-";

gotoxy(32,12);

cout<<"BUS MENU";

gotoxy(31,14);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(32,16);

cout<<"1. Create File";

gotoxy(32,18);

cout<<"2. View File";

gotoxy(32,20);

cout<<"3. Modify File";

gotoxy(32,22);

cout<<"4. Delete File";

gotoxy(31,24);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(31,26);

cout<<"Enter Command : ";

gets(Dchoice);

if(strcmp(Dchoice,"<-")==0)

return 0;

else if(strcmp(Dchoice,"1")==0)

{ bus\_createf();

goto Dloop;

}

else if(strcmp(Dchoice,"2")==0)

{ int rod=opentxtf(0,0);

if(rod==0)

goto Dloop;

else

bus\_createf();

}

else if(strcmp(Dchoice,"3")==0)

{ int mod=modifymenu();

if(mod==0)

goto Dloop;

else

bus\_createf();

}

else if(strcmp(Dchoice,"4")==0)

{ deletetxtf();

goto Dloop;

}

return 0;

}

//class customer\_rec:Manages a basic record diary as a binary file system

class customer\_rec

{ private:

char name[35],place\_from[40],place\_to[40],order\_date[10],orderf\_date[10];

char bus\_drivern[35],prline[70],choice[4];

int Bus\_no,orderno,passno;

long int phone\_no;

float netp,Tpaid,Dpaid;

char temp1[70],temp4[25],temp0[12];

long int temp2;

float temp3;

public:

char found;

customer\_rec()

{ found='n';

}

int enter\_details();

int show\_details();

char return\_found();

int return\_order();

float return\_Dpaid();

int modify();

};

int customer\_rec::enter\_details()

{ clrscr();

cout<<"<-\n"

<<"Customer Record"

<<"\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

gotoxy(1,4);

cout<<"\n Order Number : ";

cin>>orderno;

cout<<"\n Name of Customer : ";

gets(name);

cout<<"\n Phone number : ";

cin>>phone\_no;

cout<<"\nTravel from : ";

gets(place\_from);

cout<<"\nTravel to : ";

gets(place\_to);

cout<<"\nBus Number used : ";

cin>>Bus\_no;

cout<<"\n Name of bus driver hired : ";

gets(bus\_drivern);

cout<<"\n Number of Passengers : ";

cin>>passno;

cout<<"\n Order placed on : ";

gets(order\_date);

cout<<"\n Net payment : Rs ";

cin>>netp;

cout<<"\n Total amount paid : Rs";

cin>>Tpaid;

Dpaid=netp-Tpaid;

cout<<"\n Payment due : Rs "<<Dpaid;

cout<<"\n\nFullfillment of order done on : ";

gets(orderf\_date);

cout<<"\n Problem,if any, occurred during travel : ";

gets(prline);

cout<<"\n\nPress any key to go back...";

getch();

return 0;

}

int customer\_rec::show\_details()

{ clrscr();

cout<<"\n Order number : "<<orderno

<<"\n\n Name of customer : "<<name

<<"\n\n Phone number : "<<phone\_no

<<"\n\n Travel from : "<<place\_from

<<"\n\n Travel to : "<<place\_to

<<"\n\n order placed on : ";puts(order\_date);

cout<<"\n\n Bus number used : "<<Bus\_no

<<"\n\n Bus driver hired : ";puts(bus\_drivern);

cout<<"\n\n Number of Passengers : "<<passno

<<"\n\n Net payment : Rs "<<netp

<<"\n\n Total amount paid : Rs "<<Tpaid

<<"\n\n Payment due : Rs "<<Dpaid

<<"\n\n Fullfillment of order done on : "<<orderf\_date

<<"\n\n Problem,if any,occurred during travel\n : "<<prline;

cout<<"\n\n\n\t\t\t\t\tPress any key to go back";

getch();

return 0;

}

char customer\_rec::return\_found()

{ return found;

}

int customer\_rec::return\_order()

{ return orderno;

}

float customer\_rec::return\_Dpaid()

{ return Dpaid;

}

int customer\_rec::modify()

{ clrscr();

cout<<"\nEnter '.' to retain old data ";

cout<<"\n\nName of customer : "<<name<<endl;

gets(temp1);

if(strcmp(temp1,".")!=0)

strcpy(name,temp1);

cout<<"\n\n Travel from : "<<place\_from<<endl;

gets(temp1);

if(strcmp(temp1,".")!=0)

strcpy(place\_from,temp1);

cout<<"\n\n Travel to : "<<place\_to<<endl;

gets(temp1);

if(strcmp(temp1,".")!=0)

strcpy(place\_to,temp1);

cout<<"\n\n Order placed on : "<<order\_date<<endl;

gets(temp1);

if(strcmp(temp1,".")!=0)

strcpy(temp1,order\_date);

cout<<"\n\n Fullfillment of order done on : "<<orderf\_date<<endl;

gets(temp0);

if(strcmp(temp0,".")!=0)

strcpy(orderf\_date,temp0);

cout<<"\n\n Bus driver hired : "<<bus\_drivern<<endl;

gets(temp4);

if(strcmp(temp4,".")!=0)

strcpy(bus\_drivern,temp4);

cout<<"\n\n\n Enter '0' to retain old data";

cout<<"\n\n Order number : "<<orderno<<endl;

cin>>temp2;

if(temp2!=0)

orderno=temp2;

cout<<"\n\n Phone number : "<<phone\_no<<endl;

cin>>temp2;

if(temp2!=0)

phone\_no=temp2;

cout<<"\n\n Number of passengers : "<<passno<<endl;

cin>>temp2;

if(temp2!=0)

passno=temp2;

cout<<"\n\n Net payment : Rs "<<netp<<endl;

cin>>temp3;

if(temp3!=0)

netp=temp3;

cout<<"\n\n Total amount paid : Rs "<<Tpaid<<endl;

cin>>temp3;

if(temp3!=0)

Tpaid=temp3;

cout<<"\n\n Payment due : Rs "<<Dpaid<<endl;

cin>>temp3;

if(temp3!=0)

Dpaid=temp3;

return 0;

}

void customer\_menu()

{ back:

clrscr();

cout<<"<-";

char ch[4];

gotoxy(32,12);

cout<<"CUSTOMER RECORD MENU ";

gotoxy(28,13);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(29,15);

cout<<"1. Enter an order in record";

gotoxy(29,17);

cout<<"2. View an order";

gotoxy(29,19);

cout<<"3. Modify an order";

gotoxy(29,21);

cout<<"4. Delete order";

gotoxy(28,23);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

cout<<"\n\n\nEnter command : ";

gets(ch);

if(strcmp(ch,"<-")==0)

return ;

else if(strcmp(ch,"1")==0)

{ customer\_enter();

goto back;

}

else if(strcmp(ch,"2")==0)

{ customer\_view();

goto back;

}

else if(strcmp(ch,"3")==0)

{ customer\_modify();

goto back;

}

else if(strcmp(ch,"4" )==0)

{ customer\_delete();

goto back;

}

}

void customer\_enter()

{ customer\_rec b;

ofstream recdec("record.dat",ios::out|ios::binary|ios::app);

clrscr();

b.enter\_details();

clrscr();

cout<<"\nTHE DETAILS ARE : ";

b.show\_details();

recdec.write((char\*)&b,sizeof(b));

recdec.close();

return ;

}

void customer\_view()

{ clrscr();

int number,result=0;

ifstream psp("record.dat",ios::in|ios::binary);

customer\_rec bc;

cout<<" \n\n\n Enter the order number to be viewed : ";

cin>>number;

while(psp.read((char\*)&bc,sizeof(bc)))

{ int order=bc.return\_order();

if(order==number&&result!=1)

{ cout<<"\n\n\n\t\t\t\t\tFOUND!!";

delay(1000);

++result;

bc.show\_details();

}

}

if(result==0)

{ clrscr();

cout<<"\n\n\n\t\t\t\t\tNOT FOUND!!";

delay(1500);

}

return ;

}

void customer\_modify()

{ int orderi,check=0;

char chk;

retry:

clrscr();

cout<<"\n Enter order number : ";

cin>>orderi;

fstream modcus("record.dat",ios::out|ios::in|ios::binary|ios::app);

ofstream modnew("TCC.dat",ios::out|ios::binary|ios::app);

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

customer\_rec mod;

while(!modcus.eof())

{ modcus.read((char\*)&mod,sizeof(mod));

int p=mod.return\_order();

if(p==orderi&&check==0)

{ check=1;

customer\_rec mof(mod);

mof.modify();

modnew.write((char\*)&mof,sizeof(mof));

}

else if(p==orderi&&check==1)

{ cout<<" REPITITIONS HAVE BEEN REMOVED";

}

else if(p!=orderi)

{ modnew.write((char\*)&mod,sizeof(mod));

}

}

modcus.close();

modnew.close();

remove("record.dat");

rename("TCC.dat","record.dat");

if(check==0)

{ cout<<"It seems that the order you are looking for is"

<<"\n\n not in existence."

<<"\n\n Do you wish to try again?(y/n)";

cin>>chk;

if(chk=='y'||chk=='Y')

goto retry;

else

{ clrscr();

gotoxy(24,24);

cout<<" Going back to CUSTOMER RECORD MENU ";

return ;

}

}

}

void customer\_delete()

{ redel:

int deloption;

char ch,cht[4];

clrscr();

fstream wrdel("TTC.dat",ios::in|ios::out|ios::app|ios::binary);

fstream cdel("record.dat",ios::in|ios::out|ios::app|ios::binary);

customer\_rec objd;

cout<<"Enter order number to be deleted : ";

cin>>deloption;

while(cdel)

{ cdel.read((char\*)&objd,sizeof(objd));

if(objd.return\_order()==deloption)

{ cout<<"Order no. : "<<deloption<<" : Payment due : Rs."<<objd.return\_Dpaid();

cout<<"\n\n\n\n\n\n\n\t\t\tPROCEEDING TO DELETION";

delay(2000);

}

else

{ wrdel.write((char\*)&objd,sizeof(objd));

}

}

cout<<"\n\n\n\n\n\t\t\t\aORDER SUCCESFULLY DELETED";

delay(1000);

cdel.close();

wrdel.close();

remove("record.dat");

rename("TTC.dat","record.dat");

clrscr();

cout<<"\n\n\n\n\n\n\n\n\n\t\t\tPress 1 to delete another record"

<<"\n\n\t\t\tEnter '<-' to go back"

<<"\n\n Enter command : ";

gets(cht);

if(strcmp(cht,"<-")==0)

return ;

else if(strcmp(cht,"1")==0)

goto redel;

}

void menu()

{ remenu\_view:

clrscr();

int command;

gotoxy(34,12);

cout<<"TRAVEL X";

gotoxy(24,14);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(25,16);

cout<<"1. Driver Files";

gotoxy(25,18);

cout<<"2. Conductor Files";

gotoxy(25,20);

cout<<"3. Other Employee's Files";

gotoxy(25,22);

cout<<"4. Bus Files";

gotoxy(25,24);

cout<<"5. Customer's Record";

gotoxy(25,26);

cout<<"6. Exit";

gotoxy(24,28);

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

gotoxy(24,32);

cout<<"Enter Command : ";

cin>>command;

switch(command)

{ case 1: driver dr;

dr.driver\_menu();

break;

case 2: conductor cd;

cd.conductor\_menu();

break;

case 3: other\_employee oe;

oe.otemp\_menu();

break;

case 4: bus bs;

bs.bus\_menu();

break;

case 5: customer\_menu();

break;

case 6: clrscr();

gotoxy(32,24);

cout<<"Exiting the program";

delay(2000);

exit(0);

break;

default: clrscr();

gotoxy(28,24);

cout<<"\aCommand not verified";

gotoxy(28,24);

cout<<"Try Again";

delay(2000);

break ;

}

goto remenu\_view;

}

void main()

{ clrscr();

securityfunc();

getch();

}

**SNAPSHOTS OF OUTPUT SCREENS OF RUNNING PROGRAM**

1. **THE SECURITY PASSWORD FOR SECURITY OF INFORMATION**

**FOR SECURITY REASONS YOU ARE REQUIRED TO**

**ENTER THE PASSWORD TO ACCESS THE PROGRAM**

**AIBUSBIO#**

1. **THE 'HOME SCREEN'**

**TRAVEL X**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**1. Driver Files**

**2. Conductor Files**

**3. Other Employee's Files**

**4. Bus Files**

**5. Customer's Record**

**6. Exit**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Enter Command : 1**

1. **BUS DRIVER MENU. NOTE THAT THE CONDUCTOR, OTHER EMPLOYEE AND BUS FILES WORKS ON THE SAME BASIC SOURCE CODE. HENCE FILE OF ONE CATEGORY CAN BE OPERATED IN OTHER CATEGORY(EXCEPT CREATION OF FILE WHICH IS CLASS SPECIFIC)**

**<-**

**DRIVER MENU**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**1. Create File**

**2. View File**

**3. Modify File**

**4. Delete File**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Enter Command :**

1. **CREATION OF TEXT FILES**

**BANE.txt**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Enter the name of the driver : BANE**

**Enter the name of Mother of the driver : BANES**

**Enter the name of Father of the driver : BAN**

**Enter date of birth of the driver :**

**NOTE THAT:FOR EACH RECORD TO BE ENTERED A SEPARATE TEXT FILE IS CREATED**

1. **OPENING OF TEXT FILES FOR VIEW**

**praneet.txt**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**2. Name : Praneet Kapoor**

**3. Mother's Name : Sarita Kapoor**

**4. Father's Name : Nagesh kapoor**

**5. Date of Birth : 20.06.1999**

**6. Blood Group : B+**

**7. Residence Address : 1456/23,naiwala,new delhi**

**8. Aadhar Number : 1234456777**

**9. License Number : LE123 - 4r**

**10. Phone Number : 2233445**

**11. Maximum academic qualification : 12th class**

**12. Date of Joining : never**

**13. Permit : All Delhi**

**14. Criminal Record : nil**

**15. Other Details**

**16.**

**Press any key to go back**

1. **MODIFYING TEXT FILES**

**<-**

**Modifying PRANEET.txt**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Enter the lineno. of the line you wish you wish to add : 13**

**Enter the line you wish to add :**

**NOTE THAT REPLACE LINE AND ADDLINE OPTIONS DIFFER BY THEIR TREATMENT OF SELCTED LINE NO.: WHILE THE FORMER REPLACES IT, THE LATTER OPTION PUSHES IT AHEAD.**

1. **DELETING TEXT FILES**

**Enter the name of the file you wish to delete(use .txt extension) : BANE.TXT**

**1. Name : BANE**

**2. Mother's name : BANES**

**3. Father's name : BAN**

**4. Date of Birth : 30.3.1998**

**5. Blood Group : B**

**6. Residence Address : 323 D**

**7. Aadhar Number : 333454322344**

**8. License Number : 3344L**

**9. Phone Number : 33456677766**

**10. Maximum academic qualification : 3**

**11. Date of Joining : 23.03.2014**

**12. Permit : NIL**

**13. Criminal Record : NIL**

**14. Other details**

**15.**

**Are you sure that you wish to delete the file?(y/n)Y**

1. **THE CUSTOMER RECORD MENU**

**<-**

**CUSTOMER RECORD MENU**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**1. Enter an order in record**

**2. View an order**

**3. Modify an order**

**4. Delete order**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Enter command : 1**

1. **ENTERING THE RECORD OF A CUSTOMER**

**<-**

**Customer Record**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Order Number : 1**

**Name of Customer : HARSH CHAUHAN**

**Phone number : 78393893**

**Travel from : GTUTER**

**Travel to : RAOD**

**Bus Number used : 2**

**Name of bus driver hired : Fakir Singh**

**Number of Passengers : 12**

**Order placed on : 12.09.2016**

**Net payment : Rs 20000**

**Total amount paid : Rs100**

**Payment due : Rs 19900**

**Fullfillment of order done on : 16.09.2016**

**Problem,if any, occurred during travel : NIL**

**Press any key to go back...**

**NOTE THAT HERE IS ONLY ONE FILE,RECORD.DAT, WHICH STORES CUSTOMER RECORDS IN BINARY FORM**

1. **VIEWING A RECORD**

**Enter the order number to be viewed : 1**

**FOUND!!**

1. **MODIFYING RECORD**

**Enter '.' to retain old data**

**Name of customer : Bane**

**.**

**Travel from : Jhandewala**

**.**

**Travel to : Rohini**

1. **DELETING A RECORD**

**Enter order number to be deleted : 3**

**Order no. : 3 : Payment due : Rs.0**

**PROCEEDING TO DELETION**

**ORDER SUCCESFULLY DELETED**

**NOTE THAT ONLY ONE ORDER CAN BE DELETED AT A TIME**

**ADVANTAGES AND**

**DISADVANTAGES**

Every program is made to provide the best user experience. They have several features which are often showed up as advantages and some flows which are shown as disadvantages(work of critics😐)

So here are some **advantages** of this program**:**

1. It provides a very efficient platform for maintenance of file records.
2. Program directions and working is kept as simple as it could be so that it is easy to use and change/modify for user and developer respectively. Hence I don’t think that a help option was required in this program.
3. As text file have been used for majority of records, the files can be printed or viewed in notepad or Microsoft office as and when required.
4. This program can easily be changed to other transport management system☺
5. Multiple deletion of records result in ease of record management.
6. There is no hard and fast system for maintaining the cost of travel under a given circumstance. Hence the user/travel agency can easily input net travel cost as per their own convenience.
7. The use of password in the beginning of the program helps in maintaining overall security of the program.
8. The option of seeing the line number before the modification of file helps ion an accurate modification of the file as it results in reducing the chances of error by the user.

However this program does have some **disadvantages** which are listed as follows:

1. As there will be a number of text files for each individual unit/part of the organization, the system memory requirement will be high. Hence preferably a dedicated computer should be used for the use of this program.
2. A slight increase in the no. of empty lines in the end of text files after modification is observed. Hence, the user must try to enter correct records in one go.
3. Only one order can be deleted at a time.This causes multiple deletion of records as a lengthy process.
4. This program is not connected to internet and hence doesn’t have various online accessories like Google Maps,etc.
5. Improper commands may cause program to malfunction.

**BIBLIOGRAPHY\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Most of the information required in the making of this project came from the following sources :

1. Computer science with C++,Class XII,By Sumita Arora
2. www.cplusplus.com
3. notes taking during regular school classes