PROJECT WORK

Date of Submission: 21-11-23

Project Definition:

Write a C++ program to create a ticket reservation system. Program should facilitate the end user to create a separte file for each ticket record entered with ticket number as the file name. ticket record details should be :cname,tno, and destination- addr.Provide a menu driven o/p with following options for the program:

option 1->To enter ticket details and display separately each ticket file as o/p

option 2-> To edit tno of a ticket and also display the updated content to a ticket file & display appended content on screen

CodeBase:

```
#include<iostream>
#include<fstream>
#include<string.h>
#include<stdlib.h>
#include<stdio.h>
#include<cstdlib>
#include<ctime>
#include<iomanip>
#include<vector>
using namespace std;

#define RESET_CLR "\033[0m"
#define BLACK_TXT "\033[30m"
#define RED_TXT "\033[31m"
#define GREEN_TXT "\033[32m"
#define YELLOW_TXT "\033[33m"
#define BLUE_TXT "\033[34m"
#define WHITE_BG "\033[47m"

string intro = "+------------------------------------------------------------------------+\n"
   "|                    "
                  "TICKET MANAGEMENT SYSTEM "
                                        "                      |\n"
   "| Developed by: Damodar Banaulikar     (22B-CO-012)                    |\n"
   "|           : Dattaraj Gawde         (22B-CO-014)                  |\n"
   "|           : Divyam Redkar          (22B-CO-016)                  |\n"
   "|           : Ervin               (22B-CO-018)                  |\n"
   "|           : Giselle Fernandes      (22B-CO-019)                  |\n"
   "|           : Goraksh Naik           (22B-CO-020)                  |\n"
   "+----------------------------------------------------------------------+\n\n\n";

class ticket;
class BGT{
public:
   static void serach_ticket();
   static void display_details(ticket *T);
   static string generate_ticket_no();
   static string generate_train_no();
   static string get_names(int pas_count);
   static void serialize_ticket(ticket *T, fstream &ticketFile);
   static void deserialize_ticket(ticket *T, fstream &ticketFile);
   static void serialize_string(string *str, fstream &ticketFile);
   static void deserialize_string(string *str, fstream &ticketFile);
};

class ticket
{
public:
   string cname;
   string ticket_no;
   string train_no;
   string source;
   string destination;
   string coach;
   int no_pas;
   float price;

   ticket():price(0){}

   //option 1
   static void option1();
   //puts the details from ptr to file
   static int put_details_tofile(ticket *ptr);
   //option 2
   static void option2();
   //edit ticket number (tno)
   static int edit_tno_infile(ticket *old_T, ticket *new_T);
   //option 3
   static void option3();
   //add price to file
   static int add_price_tofile(ticket *ptr);
};

string BGT :: generate_ticket_no()
{
   string ticket_no;
   srand(time(0));
   // Generate a random 6-digit ticket number
   int ticket_number = rand() % 900000 + 100000;
   // Convert the ticket number to a string
   ticket_no = to_string(ticket_number);
   return ticket_no;
}
string  BGT :: generate_train_no()
{
   string train_no;
   srand(time(0));
   // Generate a random 6-digit train number
   int ticket_number = rand() % 9000 + 1000;
   // Convert the train number to a string
   train_no = to_string(ticket_number);
   return train_no;
}
string  BGT :: get_names( int pas_count )
{
   string all_names;
   string name;
   int i = 1;
   cin.ignore();
   while (i <= pas_count) {
       cout << "\nEnter passenger ["<<i++<<"] name : ";
       getline(cin, name);
       all_names += name + "$";
   }
   return all_names;
}

void BGT :: serialize_ticket(ticket *T, fstream &ticketFile)
{
   BGT::serialize_string(&T->cname, ticketFile);
   BGT::serialize_string(&T->ticket_no, ticketFile);
   BGT::serialize_string(&T->train_no, ticketFile);
   BGT::serialize_string(&T->source, ticketFile);
   BGT::serialize_string(&T->destination, ticketFile);
   BGT::serialize_string(&T->coach, ticketFile);
   ticketFile << ',';
   ticketFile << T->no_pas;
   ticketFile << ',';
   ticketFile << T->price;
}
void BGT :: deserialize_ticket(ticket *T, fstream &ticketFile)
{
   char comma;
   BGT::deserialize_string(&T->cname, ticketFile);
   BGT::deserialize_string(&T->ticket_no, ticketFile);
   BGT::deserialize_string(&T->train_no, ticketFile);
   BGT::deserialize_string(&T->source, ticketFile);
   BGT::deserialize_string(&T->destination, ticketFile);
   BGT::deserialize_string(&T->coach, ticketFile);
   ticketFile >> comma;
   ticketFile >> T->no_pas;
   ticketFile >> comma;
   ticketFile >> T->price;
}

void BGT :: serialize_string(string *str, fstream &ticketFile)
{
   ticketFile << ',';
   ticketFile << str->size();
   ticketFile << ',';
   ticketFile << str->c_str();
}
```

```
void BGT :: deserialize_string(string *str , fstream &ticketFile)
{
   int len;
   char comma;
   ticketFile >> comma;
   ticketFile >> len;
   ticketFile >> comma;
   if (ticketFile && len) {
      vector<char> tmp(len);
      ticketFile.read(tmp.data() , len);
      str->assign(tmp.data(), len);
   }
}

void BGT :: serach_ticket()
{
   ticket T;
   cout<<"Enter ticket number : ";
   cin.ignore();
   getline(cin,T.ticket_no);
   BGT::display_details(&T);
}

void BGT :: display_details(ticket *T)
{
   string ticketNo = T->ticket_no + ".txt";
   fstream ticketFile;
   ticketFile.open(ticketNo, ios::in|ios::binary);
   //checks if file exists
   ticketFile.seekg(0);
   if(ticketFile){
      BGT::deserialize_ticket(T,ticketFile);
      ticketFile.close();
   }
   else
   {
      ticketFile.close();
      cout<<RED_TXT<<"\nFile does not exist !\n"<<RESET_CLR;
      return;
   }
   istringstream iss(T->cname);
   string name;
   cout<<setfill(' ')<<"\n\n"<<WHITE_BG<<BLACK_TXT<<"+----------------------------------------------------
+"<<RESET_CLR<<endl;
   cout<<WHITE_BG<<BLACK_TXT<<"|Train Number    "<<"Number of passengers   "<<"Ticket
Number|"<<RESET_CLR<<endl;
   cout<<WHITE_BG<<BLACK_TXT<<"|"<<left<<setw(16)<<T->train_no<<setw(24)<<T-
>no_pas<<setw(12)<<T->ticket_no<<"|"<<RESET_CLR<<endl;
   cout<<WHITE_BG<<BLACK_TXT<<"|Train Coach    "<<"Boarding Station  "<<"Destination
Station|"<<RESET_CLR<<endl;
   cout<<WHITE_BG<<BLACK_TXT<<"|"<<setw(15)<<T->coach<<setw(18)<<T-
>source<<setw(19)<<T->destination<<"|"<<RESET_CLR<<endl<<"\033[2K";
   cout<<WHITE_BG<<BLACK_TXT<<"|Passenger Names :                              |"<<RESET_CLR<<endl;
   while (getline(iss, name, '$')) {
        cout<<RESET_CLR<<WHITE_BG<<BLACK_TXT<<setw(53)<<name<<"|"<<RESET_CLR<<endl;
    }
    if(T->price != 0){
        cout<<WHITE_BG<<BLACK_TXT<<"|"<<"    Price : "<<setw(40)<<T-
>price<<"|"<<RESET_CLR<<endl;
    }
    cout<<WHITE_BG<<BLACK_TXT<<"+----------------------------------------------------------------+"<<RESET_CLR;
}

void ticket :: option1()
{
    ticket T;
    cout<<"\nEnter details:\n";
    cout<<"Enter Source : ";
    cin.ignore();
    getline(cin,T.source);
    cout<<"Enter Destination : ";
    cin.ignore();
    getline(cin,T.destination);
    cout<<"Enter Train Coach (1A, 2A, 3A, CC, FC, SL, 2S) : ";
    cin.ignore();
    getline(cin,T.coach);
    cout<<"Enter number of passengers : ";
    cin>>T.no_pas;
    cout<<"Enter customer name : ";
    T.cname = BGT::get_names(T.no_pas);
    T.ticket_no = BGT::generate_ticket_no();
    cin.ignore();
    T.train_no = BGT::generate_train_no();

    if(put_details_tofile(&T) == 0){
        BGT::display_details(&T);
    }
}

int ticket :: put_details_tofile(ticket *ptr)
{
    string ticketNo = ptr->ticket_no + ".txt";

    fstream ticketFile;
    ticketFile.open(ticketNo, ios::in);
    //checks if file exists
    if(!ticketFile){
        //creates a new file if does not exist
        ticketFile.close();
        ticketFile.open(ticketNo, ios::out|ios::binary);
        BGT::serialize_ticket(ptr,ticketFile);
    }
    else{
        //gives error if file exists
        cout<<YELLOW_TXT<<"\nTicket already exists!"<<RESET_CLR;
    }
    if(ticketFile.fail()){
        cout<<YELLOW_TXT<<"\nError : I/O Error has occurred !\n"<<RESET_CLR;
        ticketFile.close();
        return 1;
    }
    if(ticketFile.bad()){
        cout<<RED_TXT<<"\nFatal Error : Invalid operation attempted !\n"<<RESET_CLR;
        ticketFile.close();
        return 1;
    }
    ticketFile.close();
    return 0;
}

void ticket :: option2()
{
    ticket old_T, new_T;
    cout<<"\nEnter ticket no to be edited : ";
    cin>>old_T.ticket_no;
    new_T.ticket_no = BGT::generate_ticket_no();
    if(ticket::edit_tno_infile(&old_T,&new_T) == 0)
        BGT::display_details(&new_T);
    getchar();
}

int ticket :: edit_tno_infile(ticket *old_T, ticket *new_T)
{
    string Ticket_old = old_T->ticket_no + ".txt";
    string Ticket_new = new_T->ticket_no + ".txt";

    //creating updated file
    fstream ticketFile;
    ticketFile.open(Ticket_old,ios::in);
    ticketFile.seekg(0);
    if(!ticketFile){
        cout<<RED_TXT<<"\nTicket with ticket number "<<old_T->ticket_no<<" does not
exist!\n"<<RESET_CLR<<endl;
    }
    else{
        // read data from file in old_T
        BGT::deserialize_ticket(old_T,ticketFile);
        old_T->ticket_no = new_T->ticket_no;

        if(ticketFile.fail()){
            cout<<YELLOW_TXT<<"\nError : I/O Error has occurred !\n"<<RESET_CLR;
            ticketFile.close();
            return 1;
        }

        ticketFile.close();

        char temp[Ticket_old.length() + 1];
        strcpy(temp, Ticket_old.c_str());

        // create new file with new name
        ticketFile.open(Ticket_new, ios::in);
        if(ticketFile){
            cout<<YELLOW_TXT<<"\nTicket already exists !\n"<<RESET_CLR;
            ticketFile.close();
            return 1;
        }
        else{
            if(remove(temp) != 0){
                cout<<RED_TXT<<"/nError in deleting old file !"<<RESET_CLR<<endl;
                return 1;
            }
            ticketFile.close();
            ticketFile.open(Ticket_new, ios::out|ios::binary);
            ticketFile.seekg(0);
            BGT::serialize_ticket(old_T,ticketFile);
        }
    }
    if(ticketFile.fail()){
        cout<<YELLOW_TXT<<"\nError : I/O Error has occurred !\n"<<RESET_CLR;
        ticketFile.close();
        return 1;
    }
    if(ticketFile.bad()){
        cout<<RED_TXT<<"\nFatal Error : Invalid operation attempted !\n"<<RESET_CLR;
        ticketFile.close();
        return 1;
    }
    if(ticketFile.good()){
        cout<<GREEN_TXT<<"\nTicket updated successfully\n"<<RESET_CLR;
    }
    ticketFile.close();
    return 0;
}

void ticket :: option3()
{
    ticket T;
    cout<<"\nEnter Ticket number : ";
    cin>>T.ticket_no;
    cin.ignore();
    cout<<"\nEnter ticket price: ";
    cin>>T.price;
    if(ticket :: add_price_tofile(&T) == 0)
        BGT::display_details(&T);
    getchar();
}

int ticket :: add_price_tofile(ticket *ptr)
{
    string ticketNo = ptr->ticket_no + ".txt";
    fstream ticketFile;
    ticket temp;

    ticketFile.open(ticketNo, ios::in|ios::binary);
    //checks if file exists
    if(ticketFile){
        //copys contents from file to temp
        ticketFile.seekg(0);
        BGT::deserialize_ticket(&temp,ticketFile);
        if(ticketFile.fail()){
            cout<<YELLOW_TXT<<"\nError : I/O Error has occurred !\n"<<RESET_CLR;
            ticketFile.close();
            return 1;
        }

        ticketFile.close();
        //update price in temp
        temp.price = ptr->price;
        //update contents of file
        ticketFile.open(ticketNo, ios::out|ios::binary);
        ticketFile.seekg(0);
        BGT::serialize_ticket(&temp, ticketFile);
        if(ticketFile.fail()){
            cout<<YELLOW_TXT<<"\nError : I/O Error has occurred !\n"<<RESET_CLR;
            ticketFile.close();
            return 1;
        }
    }
    else{
        //gives error if file does not exists
        cout<<RED_TXT<<"\nFile does not exists!"<<RESET_CLR<<endl;
        ticketFile.close();
        return 1;
    }
    if(ticketFile.bad()){
        cout<<RED_TXT<<"\nFatal Error : Invalid operation attempted !\n"<<RESET_CLR;
        ticketFile.close();
        return 1;
    }
    ticketFile.close();
    return 0;
}

int main()
{
    int choice;

    while (1)
    {
        system("CLS");
        cout<<intro;
        cout <<BLUE_TXT<<"\n-----------------------MENU----------------------------" << endl;
        cout << "[1] Enter Ticket Details " << endl;
        cout << "[2] Edit Ticket Number " << endl;
        cout << "[3] Append Price to a Ticket " << endl;
        cout << "[4] Search and Display Ticket"<<endl;
        cout << "[5] Exit" << endl;
        cout << "Enter your choice: "<<RESET_CLR;
        cin >> choice;
        switch (choice)
        {
        case 1:
            ticket::option1();
            break;

        case 2:
            ticket::option2();
            break;

        case 3:
            ticket::option3();
            break;

        case 4:
            BGT::serach_ticket();
            break;
        case 5:
            return 0;

        default:
            cout <<YELLOW_TXT<< "Invalid choice. Please try again."<<RESET_CLR<< endl;
            break;
        }
        cout<<GREEN_TXT<<"\ncontinue ? (y/n) :"<<RESET_CLR;
        char c = getchar();
        if(c =='n' || c == 'N')
            return 0;
    }
    return 0;
}
```

Code Documentation:

BGT Class:

1. search_ticket()

Purpose:
Searches for a ticket based on the provided ticket number and displays its details.

Input Parameters:
None

Output Values:
None

Exceptions:
None

Example Usage:
BGT::search_ticket();

2. display_details(ticket *T)

Purpose:
Displays details of a ticket.

Input Parameters:
- T: Pointer to a ticket object.

Output Values:
None

Exceptions:
None

Example Usage:
ticket T;
// Populate T with ticket details
BGT::display_details(&T);

3. generate_ticket_no()

Purpose:
Generates a random 6-digit ticket number.

Input Parameters:
None

Output Values:
- Returns a string representing the generated ticket number.

Exceptions:
None

Example Usage:
string ticket_no = BGT::generate_ticket_no();

4. generate_train_no()

Purpose:
Generates a random 6-digit train number.

Input Parameters:
None

Output Values:
- Returns a string representing the generated train number.

Exceptions:
None

Example Usage:
string train_no = BGT::generate_train_no();

5. get_names(int pas_count)

Purpose:
Gets names of passengers from user input.

Input Parameters:
- pas_count: Number of passengers.

Output Values:
- Returns a string containing names separated by '$' delimiter.

Exceptions:
None

Example Usage:
int passenger_count = 3;
string all_names =
BGT::get_names(passenger_count);

6. serialize_ticket(ticket *T, fstream &ticketFile)

Purpose:
Serializes ticket details to a file.

Input Parameters:

- T: Pointer to a ticket object.
- ticketFile: Reference to a file stream.

Output Values:
None

Exceptions:
None

Example Usage:

```
ticket T;
fstream ticketFile;
// Open ticket file stream
BGT::serialize_ticket(&T, ticketFile);
```

7. deserialize_ticket(ticket *T, fstream &ticketFile)

Purpose:
Deserializes ticket details from a file.

Input Parameters:
- T: Pointer to a ticket object.
- ticketFile: Reference to a file stream.

Output Values:
None

Exceptions:
None

Example Usage:

```
ticket T;
fstream ticketFile;
// Open ticket file stream
BGT::deserialize_ticket(&T, ticketFile);
```

8. serialize_string(string *str, fstream &ticketFile)

Purpose:
Serializes a string to a file.

Input Parameters:
- str: Pointer to a string object.
- ticketFile: Reference to a file stream.

Output Values:
None

Exceptions:
None

Example Usage:

```
string myString = "Hello, World!";
fstream ticketFile;
// Open ticket file stream
BGT::serialize_string(&myString, ticketFile);
```

9. deserialize_string(string *str, fstream &ticketFile)

Purpose:
Deserializes a string from a file.

Input Parameters:
- str: Pointer to a string object.
- ticketFile: Reference to a file stream.

Output Values:
None

Exceptions:
None

Example Usage:

```
string myString;
fstream ticketFile;
// Open ticket file stream
BGT::deserialize_string(&myString, ticketFile);
```

ticket Class:

1. option1()

Purpose:
Allows the user to enter details for a new ticket.

Input Parameters:
None

Output Values:
None

Exceptions:
None

Example Usage:

```
ticket::option1();
```

2. put_details_tofile(ticket *ptr)

Purpose:
Puts ticket details into a file.

Input Parameters:
- ptr: Pointer to a ticket object.

Output Values:
- Returns 0 if successful.

Exceptions:
- Displays an error message if the ticket already exists.

Example Usage:
ticket T;
// Populate T with ticket details
int result = ticket::put_details_tofile(&T);

3. option2()

Purpose:
Allows the user to edit a ticket number in the file.

Input Parameters:
None

Output Values:
None

Exceptions:
None

Example Usage:
ticket::option2();

4. edit_tno_infile(ticket *old_T, ticket *new_T)

Purpose:
Edits the ticket number in the file.

Input Parameters:
- old_T: Pointer to a ticket object with the old ticket number.
- new_T: Pointer to a ticket object with the new ticket number.

Output Values:
- Returns 0 if successful.

Exceptions:
- Displays an error message if the old ticket number does not exist.

Example Usage:
ticket oldTicket, newTicket;
// Populate oldTicket and newTicket with details
int result = ticket::edit_tno_infile(&oldTicket, &newTicket);

5. option3()

Purpose:
Allows the user to add a price to an existing ticket.

Input Parameters:
None

Output Values:
None

Exceptions:
None

Example Usage:
ticket::option3();

6. add_price_tofile(ticket *ptr)

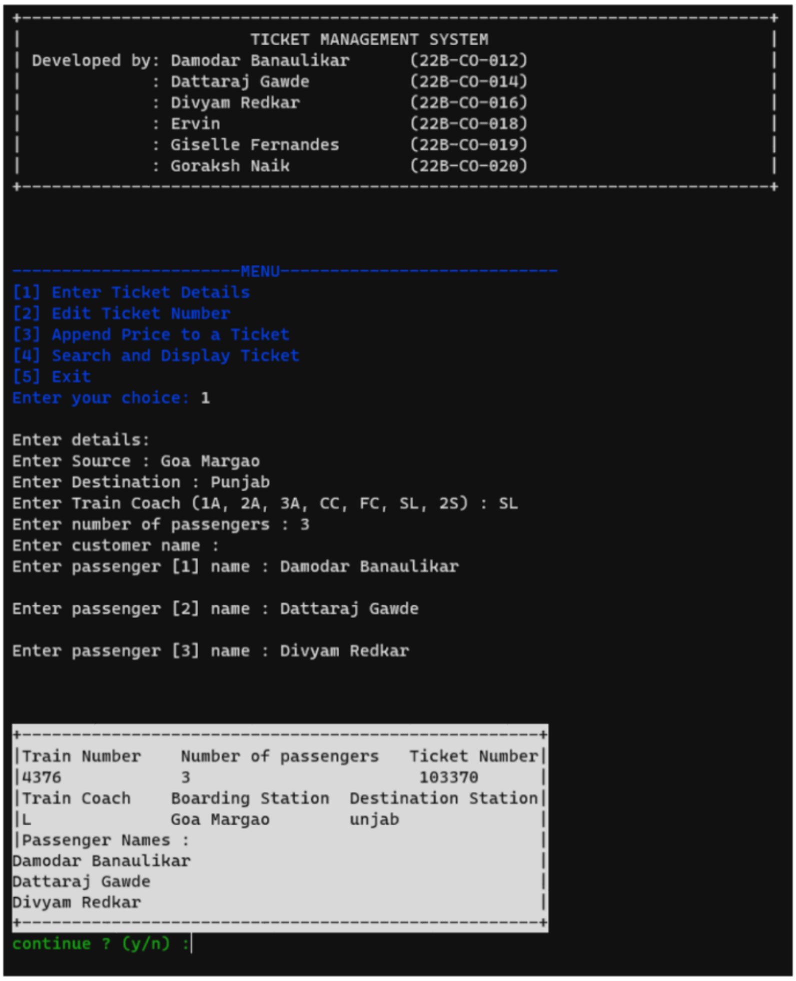
Purpose:
Adds a price to an existing ticket in the file.

Input Parameters:
- ptr: Pointer to a ticket object with the ticket number and price.

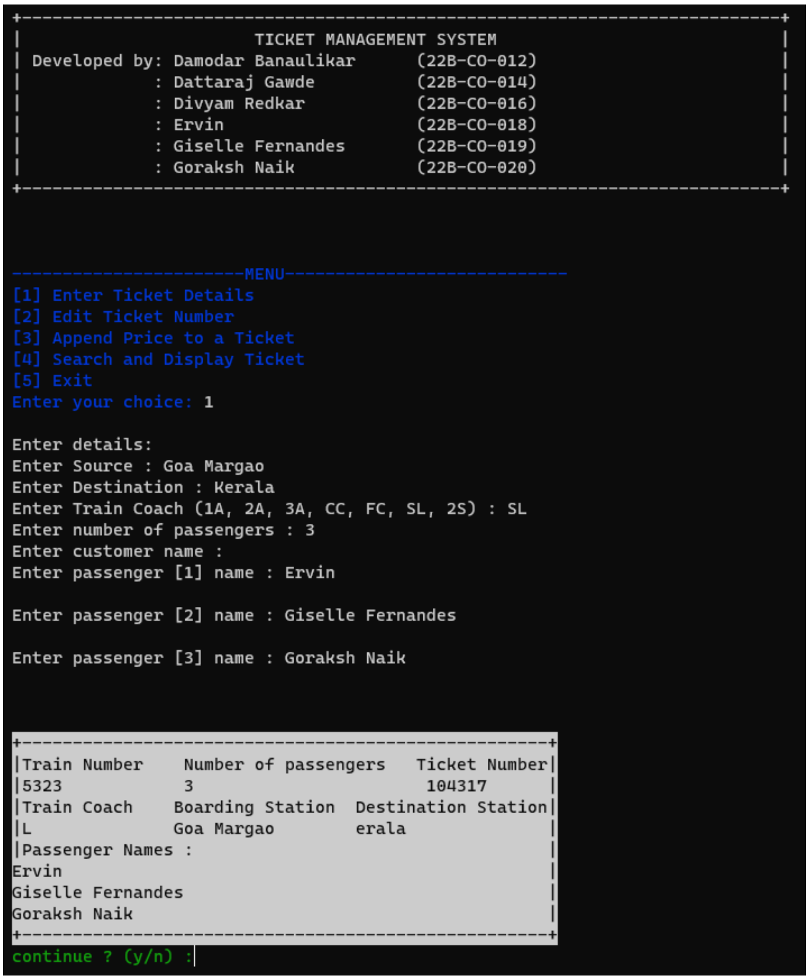
Output Values:
- Returns 0 if successful.

Exceptions:
- Displays an error message if the ticket does not exist.

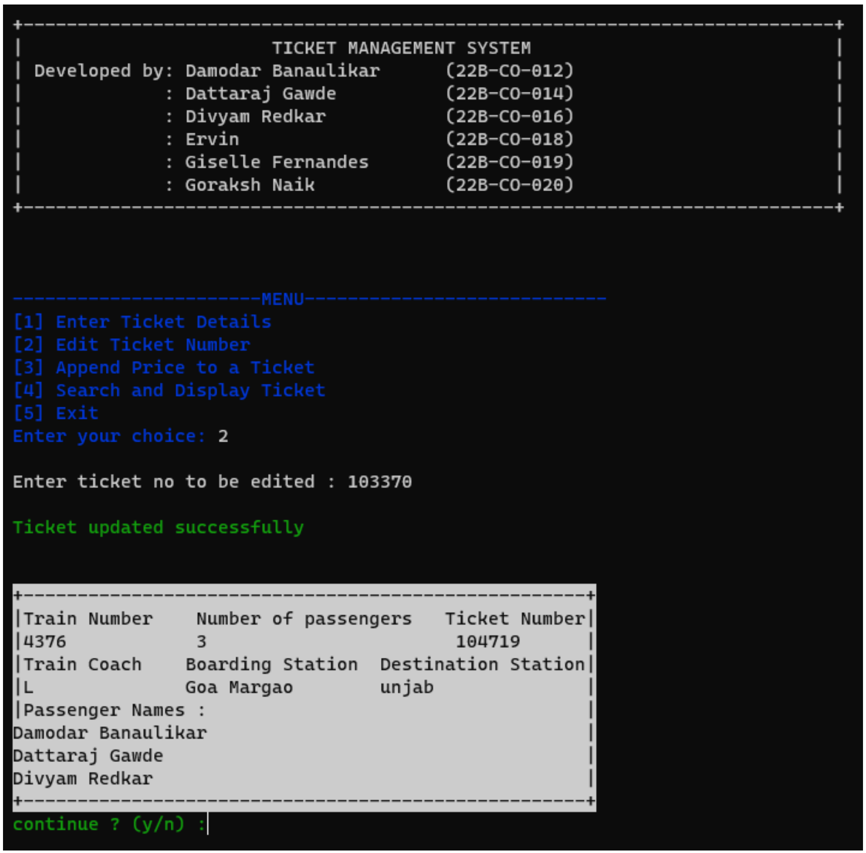
Example Usage:
ticket T;
// Populate T with ticket details
int result = ticket::add_price_tofile(&T);



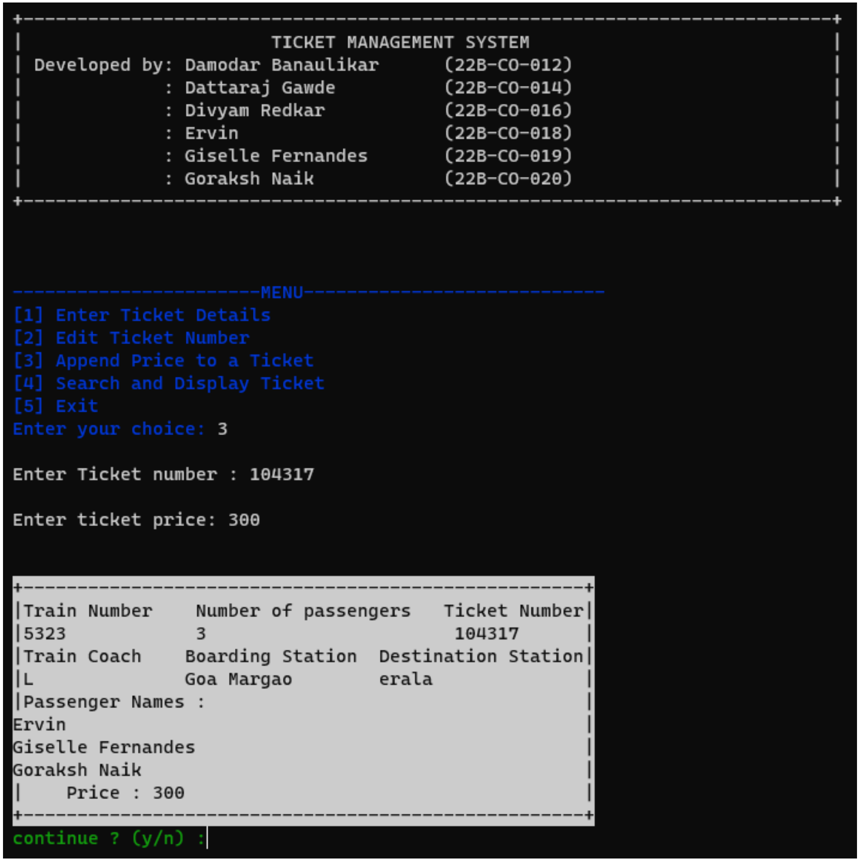
```
+-----------------------------------------------------------------------------+
|                       TICKET MANAGEMENT SYSTEM                              |
| Developed by: Damodar Banaulikar      (22B-CO-012)                          |
|             : Dattaraj Gawde          (22B-CO-014)                          |
|             : Divyam Redkar           (22B-CO-016)                          |
|             : Ervin                   (22B-CO-018)                          |
|             : Giselle Fernandes       (22B-CO-019)                          |
|             : Goraksh Naik            (22B-CO-020)                          |
+-----------------------------------------------------------------------------+



-----------------------MENU---------------------------
[1] Enter Ticket Details
[2] Edit Ticket Number
[3] Append Price to a Ticket
[4] Search and Display Ticket
[5] Exit
Enter your choice: 1

Enter details:
Enter Source : Goa Margao
Enter Destination : Punjab
Enter Train Coach (1A, 2A, 3A, CC, FC, SL, 2S) : SL
Enter number of passengers : 3
Enter customer name :
Enter passenger [1] name : Damodar Banaulikar

Enter passenger [2] name : Dattaraj Gawde

Enter passenger [3] name : Divyam Redkar



+-----------------------------------------------------+
|Train Number    Number of passengers   Ticket Number|
|4376            3                      103370       |
|Train Coach     Boarding Station   Destination Station|
|L               Goa Margao         unjab            |
|Passenger Names :                                   |
Damodar Banaulikar                                   |
Dattaraj Gawde                                       |
Divyam Redkar                                        |
+-----------------------------------------------------+
continue ? (y/n) :
```



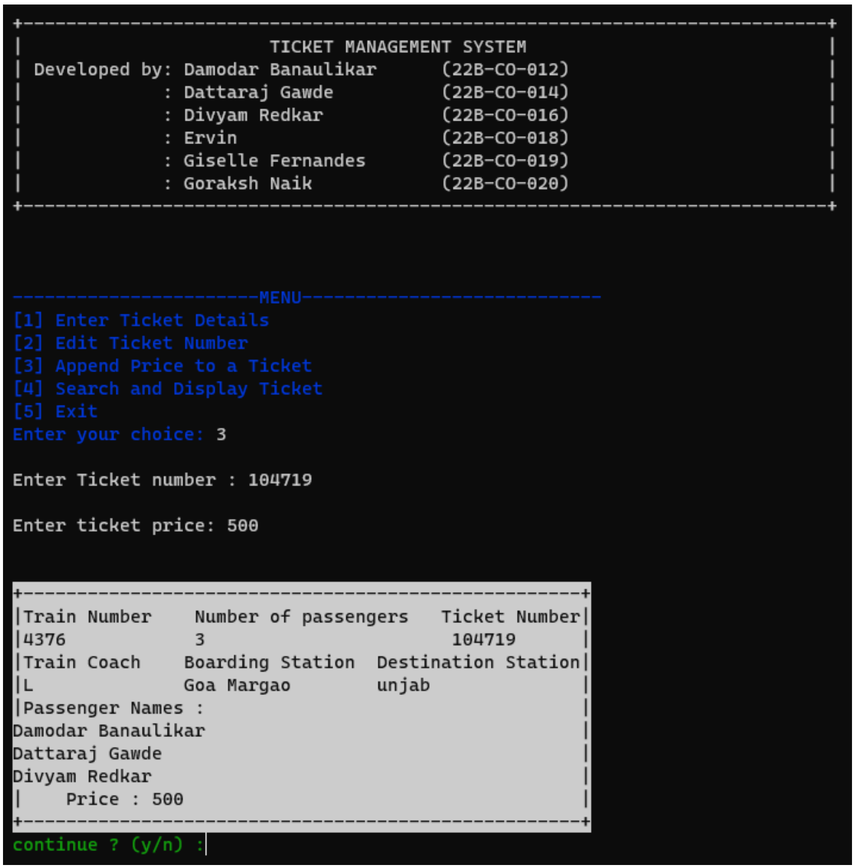
```
+-----------------------------------------------------------------------------+
|                       TICKET MANAGEMENT SYSTEM                              |
| Developed by: Damodar Banaulikar      (22B-CO-012)                          |
|             : Dattaraj Gawde          (22B-CO-014)                          |
|             : Divyam Redkar           (22B-CO-016)                          |
|             : Ervin                   (22B-CO-018)                          |
|             : Giselle Fernandes       (22B-CO-019)                          |
|             : Goraksh Naik            (22B-CO-020)                          |
+-----------------------------------------------------------------------------+



-----------------------MENU---------------------------
[1] Enter Ticket Details
[2] Edit Ticket Number
[3] Append Price to a Ticket
[4] Search and Display Ticket
[5] Exit
Enter your choice: 1

Enter details:
Enter Source : Goa Margao
Enter Destination : Kerala
Enter Train Coach (1A, 2A, 3A, CC, FC, SL, 2S) : SL
Enter number of passengers : 3
Enter customer name :
Enter passenger [1] name : Ervin

Enter passenger [2] name : Giselle Fernandes

Enter passenger [3] name : Goraksh Naik



+-----------------------------------------------------+
|Train Number    Number of passengers   Ticket Number|
|5323            3                      104317       |
|Train Coach     Boarding Station   Destination Station|
|L               Goa Margao         erala            |
|Passenger Names :                                   |
Ervin                                                |
Giselle Fernandes                                    |
Goraksh Naik                                         |
+-----------------------------------------------------+
continue ? (y/n) :
```



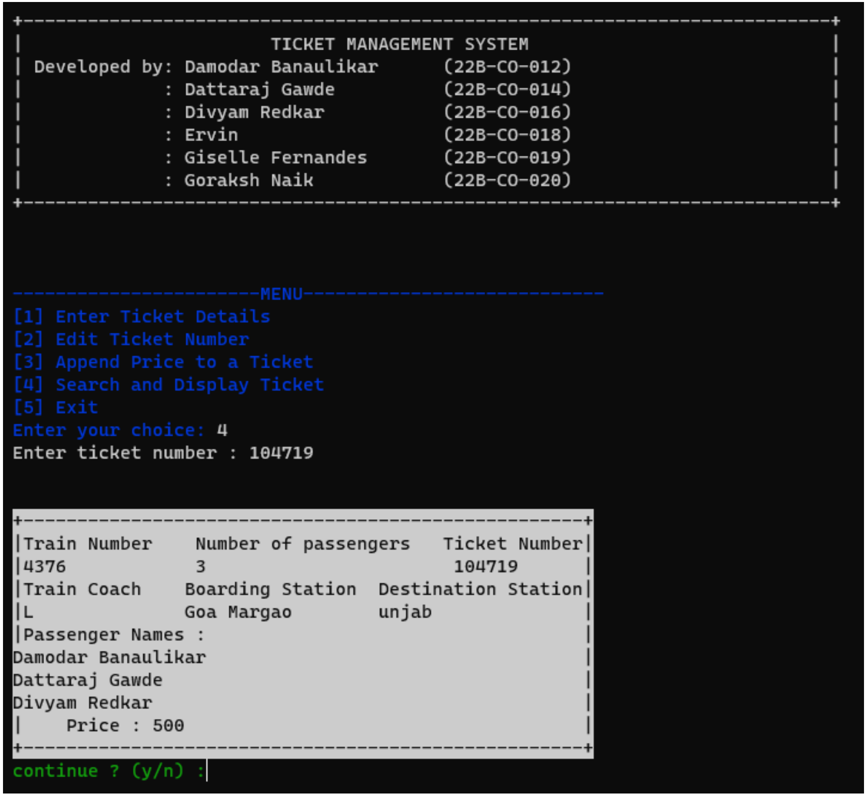
```
+-----------------------------------------------------------------------------+
|                       TICKET MANAGEMENT SYSTEM                              |
| Developed by: Damodar Banaulikar      (22B-CO-012)                          |
|             : Dattaraj Gawde          (22B-CO-014)                          |
|             : Divyam Redkar           (22B-CO-016)                          |
|             : Ervin                   (22B-CO-018)                          |
|             : Giselle Fernandes       (22B-CO-019)                          |
|             : Goraksh Naik            (22B-CO-020)                          |
+-----------------------------------------------------------------------------+



-----------------------MENU----------------------------
[1] Enter Ticket Details
[2] Edit Ticket Number
[3] Append Price to a Ticket
[4] Search and Display Ticket
[5] Exit
Enter your choice: 2

Enter ticket no to be edited : 103370

Ticket updated successfully


+------------------------------------------------------+
|Train Number    Number of passengers   Ticket Number|
|4376            3                        104719     |
|Train Coach    Boarding Station  Destination Station|
|L              Goa Margao        unjab              |
|Passenger Names :                                   |
Damodar Banaulikar                                   |
Dattaraj Gawde                                       |
Divyam Redkar                                        |
+------------------------------------------------------+
continue ? (y/n) :
```



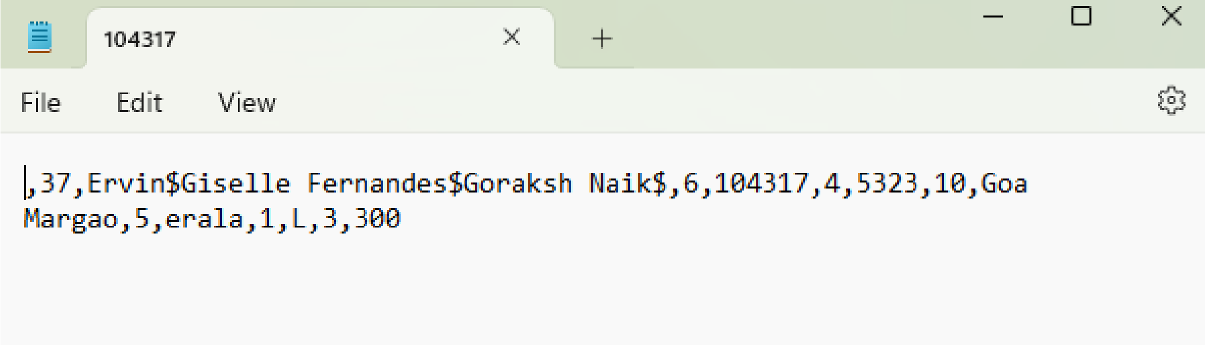
```
+-----------------------------------------------------------------------------+
|                       TICKET MANAGEMENT SYSTEM                              |
| Developed by: Damodar Banaulikar      (22B-CO-012)                          |
|             : Dattaraj Gawde          (22B-CO-014)                          |
|             : Divyam Redkar           (22B-CO-016)                          |
|             : Ervin                   (22B-CO-018)                          |
|             : Giselle Fernandes       (22B-CO-019)                          |
|             : Goraksh Naik            (22B-CO-020)                          |
+-----------------------------------------------------------------------------+



-----------------------MENU----------------------------
[1] Enter Ticket Details
[2] Edit Ticket Number
[3] Append Price to a Ticket
[4] Search and Display Ticket
[5] Exit
Enter your choice: 3

Enter Ticket number : 104317

Enter ticket price: 300


+------------------------------------------------------+
|Train Number    Number of passengers   Ticket Number|
|5323            3                        104317     |
|Train Coach    Boarding Station  Destination Station|
|L              Goa Margao        erala              |
|Passenger Names :                                   |
Ervin                                                |
Giselle Fernandes                                    |
Goraksh Naik                                         |
|    Price : 300                                     |
+------------------------------------------------------+
continue ? (y/n) :
```



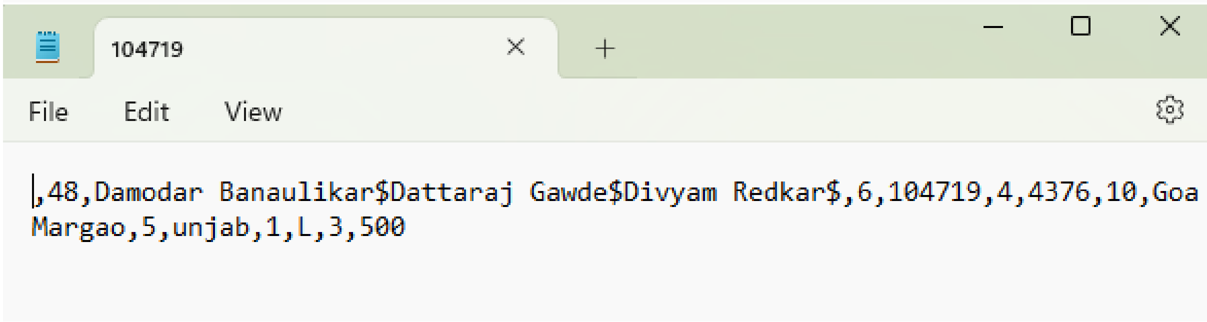
```
+----------------------------------------------------------------------------+
|                        TICKET MANAGEMENT SYSTEM                            |
| Developed by: Damodar Banaulikar      (22B-CO-012)                         |
|             : Dattaraj Gawde          (22B-CO-014)                         |
|             : Divyam Redkar           (22B-CO-016)                         |
|             : Ervin                   (22B-CO-018)                         |
|             : Giselle Fernandes       (22B-CO-019)                         |
|             : Goraksh Naik            (22B-CO-020)                         |
+----------------------------------------------------------------------------+



-----------------------MENU----------------------------
[1] Enter Ticket Details
[2] Edit Ticket Number
[3] Append Price to a Ticket
[4] Search and Display Ticket
[5] Exit
Enter your choice: 3

Enter Ticket number : 104719

Enter ticket price: 500


+-----------------------------------------------------+
|Train Number    Number of passengers   Ticket Number|
|4376            3                         104719    |
|Train Coach     Boarding Station  Destination Station|
|L               Goa Margao        unjab              |
|Passenger Names :                                    |
Damodar Banaulikar                                    |
Dattaraj Gawde                                        |
Divyam Redkar                                         |
|    Price : 500                                      |
+-----------------------------------------------------+
continue ? (y/n) :
```



```
+----------------------------------------------------------------------------+
|                        TICKET MANAGEMENT SYSTEM                            |
| Developed by: Damodar Banaulikar      (22B-CO-012)                         |
|             : Dattaraj Gawde          (22B-CO-014)                         |
|             : Divyam Redkar           (22B-CO-016)                         |
|             : Ervin                   (22B-CO-018)                         |
|             : Giselle Fernandes       (22B-CO-019)                         |
|             : Goraksh Naik            (22B-CO-020)                         |
+----------------------------------------------------------------------------+



-----------------------MENU----------------------------
[1] Enter Ticket Details
[2] Edit Ticket Number
[3] Append Price to a Ticket
[4] Search and Display Ticket
[5] Exit
Enter your choice: 4
Enter ticket number : 104719


+-----------------------------------------------------+
|Train Number    Number of passengers   Ticket Number|
|4376            3                         104719    |
|Train Coach     Boarding Station  Destination Station|
|L               Goa Margao        unjab              |
|Passenger Names :                                    |
Damodar Banaulikar                                    |
Dattaraj Gawde                                        |
Divyam Redkar                                         |
|    Price : 500                                      |
+-----------------------------------------------------+
continue ? (y/n) :
```



```
,37,Ervin$Giselle Fernandes$Goraksh Naik$,6,104317,4,5323,10,Goa
Margao,5,erala,1,L,3,300
```



```
,48,Damodar Banaulikar$Dattaraj Gawde$Divyam Redkar$,6,104719,4,4376,10,Goa
Margao,5,unjab,1,L,3,500
```

Developed by

: Damodar Banaulikar	(22B-CO-012)
: Dattaraj Gawde	(22B-CO-014)
: Divyam Redkar	(22B-CO-016)
: Ervin	(22B-CO-018)
: Giselle Fernandes	(22B-CO-019)
: Goraksh Naik	(22B-CO-020)

FACULTY: - Prof. AMIT PATIL **CLASS: - S.E COMPUTER**

Prof. NITESH NAIK