**ABSTRACT**

Airline reservation System is a computerized system used to store and retrieve information and conduct transactions related to air travel. The project is aimed apex posing the relevance and importance of Airline Reservation Systems. It is project extol wards enhancing the relationship between customers and airline agencies through the use of ars and thereby making it convenient for the customers to book the flights as when they require such that they can utilize this software to make reservations.

This software has two parts. First is user part and the administrator part. Use part is used as a front end and administrator is the back end. Administrator is used by airline authority. It will allow the customers to access database and allow new customers to sign up for online access.

After search the system display list of available flights and allows customer to choose a particular flight. Then the system checks for the availability of seats on the flight. If the seats are available then the system allows the passenger to book a seat. Otherwise it asks the user to choose another flight.

To book a flight the system asks the customer to enter his details such as name, address, city, state, and credit card number and contact number. Then it checks the validity of card and book the flight and update the airline database and user database. The system also allows the customer to cancel his/her reservation, if any problem occurs.

The main purpose of this software is to reduce the manual errors involved in the airline reservation process and make it convenient for the customers to book the flights as when they require such that they can utilize this software to make reservations, modify reservations or cancel a particular reservation.

#### INTRODUCTION

The project of “Airline Reservation system “ is going to be developed for PU school of science and technology PUSAT , Biratnagar, provence -01, Nepal. This project which has been built using C++ helps users to make easy flight ticket reservations. This project has which has been built using C++ to help users to make flight ticket reservations.

It allows the using to perform for function:-

a. Booking a flight

b. Cancel a flight

c. Check the booked tickets

d. Exit the program

The user can book both domestic and international flight tickets by entering the required details of the journey. A PNR number is generated once the ticket is booked. This number is highly essential in checking the booked ticket and also in cancelling the same. File handling concept is used to satisfy this purpose. Then **airline reservation system project in C++ with output** displays the reservation number. Similarly, if a user wishes to change a reservation, he or she must include the current seat number and choose a new seat number. The system makes canceling a reservation simple; all the user has to do is include their reservation number.

Airlines are classified by their router and by their schedules. The two major classification are domestic airlines and internationals airlines .  Domestic airlines provide services within a country. International airlines, on the other hand, operate both within a nation and between two or more nations. The International Civil Aviation Organization also distinguishes between scheduled and nonscheduled operators. Nonscheduled airlines offer either domestic or international air transportation services, but they do not maintain definite departure or arrival times. Many individual countries have established various classes of air carriers. These categories often include local service airlines, trunk airlines that service large hubs, international airlines, all-cargo lines, and charter services.

**Objective of Airlines Reservation System**

The main objective of the Airlines of the Reservation system is to manage the details of Airlines. Ticket , Flights , Customer , Booking counter , Venders. It manages all the information about Airlines Ticket , Bookings ,Venders , Airlines Ticket. The project is totally built at administrative end and thus only the administrator is guaranteed the access . The purpose of the project is to build an application program to reduce the manual work for managing the airlines Ticket , Flights , Booking , Customer. It tracks all the details about the customer , Booking counter , venders.

**Features of Airlines Reservation System are as follows:-**

* Provides the searching facilities based on various factors. Such as Airlines Ticket , Customer , Booking Counter ,Venders
* The transactions are executed in offline mode online data for Airlines Ticket , Flights capture and modification is not possible.
* It tracks all the information of flights , Booking , Booking counter etc.
* Manage the information of Flights
* Show the information and description of the Airlines Ticket , Customer.
* It generates the report on Airlines Ticket , Flights Bookings.
* Provide filter reports on Customer Booking counter venders
* Manage the information of Airlines Ticket.
* Manage the information of Booking counter.
* Integration of all records of venders.

**TABLE OF CONTENTS**

**CHAPTER TITLE PAGE NO.**

**ABSTRACT** 1

**INTRODUCTION 2-3**

**LIST OF TABLE 4-5**

1. **INTRODUCTION**  6-7

1.1 Overview of the project

1.2 Purpose of the Project

1.3 Problem definition

1.4 Project plan and scope of the project

2**. SYSTEM ANALYSIS 8-10**

2.1 General

2.2 Existing system

2.2.1 Drawbacks of existing system

2.3 Proposed system

2.3.1 Advantages of proposed system

2.4 Feasibility study

2.4.1 Technical feasibility

3. **SYSTEM REQUIREMENTS 11**

3.1 Hardware requirements

3.2 Software requirements

3.2.1 Software descriptions: Frontend

3.2.2 Software descriptions: Backend

4. 3.2.3 Features of VB and ORACLE 11g

4. **SYSTEM DESIGN 12-13**

4.1 Introduction to ER Model

4.2 Introduction to ER Diagram

4.2.1 Description of the symbol

4.2.2 Description of the Entities

4.2.3 ER-Diagram (your project)

5. **SYSTEM IMPLEMENTATION 14**

5.1 Module Identification

5.2 Module description

5.1 Flight Details

5.2.2 Module 2: Check Availability

3: Book Ticket

4: Exit

6**. CONCLUSION AND REFERENCES 15**

7. **CODING 16-47**

Screenshots

REFERENCES

**CHAPTER 1**

**INTRODUCTION**

**1.INTRODUCTION**

The project aims to design an Airline Reservation System application which enables the customers to search and book flight . Airline Reservations Systems contain fare tariffs, passenger reservations and ticket records

**1.1 OVERVIEW OF THE PROJECT**

The Airline Reservation System project is an implementation of a general Airline Ticketing website like Orbit, which helps the customers to search the availability of flights, book and cancel the flight tickets. This project also covers adding, deleting or modifying the customer details and flights. In general, this website would be designed to perform like any other airline ticketing website available online.

**1.2 PURPOSE OF THE PROJECT**

The purpose of this project is to implement or to design a database for an airline reservation system to check the flight details ,book and cancel flight tickets. It makes the process of booking and cancelling flight tickets simple and easy for the passengers.

**1.3 PROBLEM DEFINITION**

Normally a person wants to reserve his ticket and he has to contact at nearest Overseas Travels branch. The Airline Reservation System provides an interface to schedule flights and reservations for an airline through internet. Its responsibility is to keep track of system users, customers, Airbus information, flight information and cancellation.

**1.4 PROJECT PLAN AND SCOPE OF THE PROJECT**

Airline Reservation System is one the modifications that were carried out in the Passenger Service System so that the working and availability of Service area can be broadened. On one hand, it helps the customers and on the other, it also makes the life of the airline service companies easier by keeping all the records of the passengers and if there is any change in the fight due to some reason, the passengers are promptly informed. This system is also used by companies to keep track of user preferences of regular travelers so that they can provide better service and give offers to customers.

**CHAPTER 2**

**SYSTEM ANALYSIS**

**2. SYSTEM ANALYSIS**

Systems analysis is the study of sets of interacting entities, including computer systems analysis. This field is closely related to requirement analysis or operations research.

It is also "an explicit formal inquiry carried out to help someone (referred to as the decision maker). It identifies a better course of action and make a better decision than he might otherwise have made."

The development of a computer-based information system includes a systems analysis phase which produces or enhances datamodel which itself is a precursor to creating or enhancing a database.

There are a number of different approaches to system analysis. When a computer-based information system is developed, systems analysis would constitute the following steps:-

* The development of a feasibility study, involving determining whether a project is economically, socially, technologically and organizationally feasible.
* Conducting fact-finding measures, designed to ascertain the requirements of the system's end-users. These typically span interviews, questionnaires, or visual observations of work on the existing system.

**2.1 GENERAL**

Systems analysis researchers apply methodology to the analysis of systems involved to form an database for checking their results. System analysis is used in every field where there is a work of developing something. Analysis can also be defined as a series of components that perform organic function together.

**2.2 EXISTING SYSTEM**

The existing system is that the passenger must fill up the data manually and must submit it to the reservation counter. It may take a lot of time to process it and to book the flight. Therefore there is wastage of time. Since the data is entered manually, the probability of error or mistakes is high

**2.2.1 DRAWBACKS OF EXISTING SYSTEM**

**Entering Record:-**

Entry of each record is done manually each time the record is done manual each time the record is maintained on paper and it maximizes the maintenance of additional files.

**Searching the record:-**

Due to absence of unique identification of a flight, the searching of record takes much time and increases the time wastage.

**Deleting the Record**

In the current system the concept of deleting record is tedious.

**Modification of Records:-**

If any modification is required it is done directly on the documents being preserved in correspondence to account information.

**2.3 PROPOSED SYSTEM**

To avoid the limitation of current system it’s necessary to design and develop a new system which have the following benefit and the existing system.

1. Everything is automated which reduce the risk factor.

2. Flexibility in generating of information. 3. Quick retrieved and maintenance of data.

4. Highly accurate.

5. User satisfaction

**2.3.1 ADVANTAGES OF PROPOSED SYSTEM**

* The proposed system due to computerized is much faster in reservation process, cancellation process and transactions.

* Transfer of information from various branches would become easier and faster.
* Managing and maintaining data becomes easier and cost effective due to very high amount and reliability of storage space available in the proposed system.
* Customer services can not only be satisfied but also enhanced to the extent that one can obtain or cancel a reservation from any given time.

**2.4 FEASIBILITY STUDY**

Feasibility study is a report directed management. It evaluates the impact of the proposed changes in the area(s) in question. The report is a formal document for management, brief enough and sufficiently, non technical to be understandable, yet detailed enough to provide the basis for system design.

**2.4.1 Technical feasibility**

Technical feasibility centers around the existing system (hardware, software, etc) into what it can sort the proposed addition.

**CHAPTER 3**

**SYSTEM REQUIREMENTS**

**3.1 HARDWARE REQUIREMENT**

An Intel based central processing unit capable of running any sort of windows operating system such as Pentium based workstation.

1. Minimum 64 MB RAM (128 MB Desirable) at server.

2. Minimum 60 MB of free disk space for files.

3. A CD Rom drive 4. Minimum 48 MB of RAM at workstation.

5. VGA 15” colours monitor for workstation.

**3.2 SOFTWARE REQUIREMENT:-**

1. Windows 10, and 11

2. VB 6.0

3. The software requirements are as follows.

Microsoft word

4. ORACLE 11g

5. Codeblock 20.3

**3.2.1 SOFTWARE FRONT END:-**

. The front end for airline reservation system is Visual Basic

**3.2.2 SOFTWARE BACK END:-**

ORACLE is the back end used to create the tables for airline reservation systems

**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 INTRODUCTION TO E-R MODEL:-**

An ER model is an abstract way to describe a database. Describing a database usually starts with a relational database, which stores data in tables. Some of the data in these tables point to data in other tables - for instance, your entry in the database could point to several entries for each of the phone numbers that are yours. The ER model would say that you are an entity, and each phone number is an entity, and the relationship between you and the phone numbers is 'has a phone number'. Diagrams created to design these entities and relationships are called entity– relationship diagrams or ER diagrams.

**4.2 INTODUCTION TO E-R DIAGRAM:-**

Repre Represents table name

Represent attribute of a table

Represents relationship

Fig no.4.1 Description of tables

**4.2.2 DESCRIPTION OF ENTITIES**

Flightdata- this table contains all the information about the flight such as flight type, PNR number, source, destination, branch code, service code and class.

Passenger- this table contains all the details about the passenger like name, address, passport number, and status of the passenger.

**4.2.3 ER-DIAGRAM**

Flight Access

Passenger

Flight

Fig:- 4.2 ER- diagram for Airline Reservation System

**CHAPTER 5**

**SYSTEM IMPLEMENTATION**

**5.1 MODULE IDENTIFICATION**

Each form in Microsoft Visual Basic 6.0 is the module of our project. We are using about 8 forms in our project. Each form has a unique purpose. In 5 modules or database will be displayed that describes the information about all the attributes of the tables that we have created in Oracle11g.

**5.2 MODULE DESCRIPTION**

The tickets issued should have the details such as plane number, ticket number, seat number, traveler’s name, time of departure. The traveler should be informed about the check-in time. The names of the fields involved in the airline reservation system are given below :-

* FLIGHT DETAILS
* CHECK AVAILABILITY
* BOOK TICKET
* EXIT

**5.2.1 MODULE 1:FLIGHT DETAILS**

This module is used to view the flight details with ease and it tends the passenger to book tickets without much difficulty.

**5.2.2MODULE 2:CHECK AVAILABILITY**

This module is used to check the availability of the flights and the information of the seats in that flight.

**5.2.3 MODULE 3:BOOK TICKET**

This module is used to book the ticket after checking the availability of tickets in the flights. A ticket can be booked just by entering the passenger name, age and their details.

**5.2.4 MODULE 4:EXIT**

This module is used to exit from the reservation form.

**CHAPTER 6**

**CONCLUSION AND FUTURE ENHANACEMENT**

**6. CONCLUSION AND FUTURE ENHANCEMENT**

The Airline reservation system has been a way of minimizing the clerical work, which is almost a routine and consumes the most precious time. This AIRLINE RESERVATION SYSTEM has been an attempt to help the user to minimize his workload along with minimizing the paper works and saving of time .The system has been developed in a way to make it very user friendly. It provides an on-line message and an error detection and error messages every time the user needs. Any person having a little bit of window based can run this system without any pain. As a FUTURE ENHANCEMENT we have decided to further enhance with a seat reservation available. It is to fulfill passengers request to sit where they prefer. They are allowed to choose their seat whether near to window’s seat or in the middle.

**Coding of AIRLINE RESERVATION SYSTEM**

include <iostream>

#include<conio.h>

#include<fstream>

#include<string.h>

using namespace std;

int dom\_pnr=100;

int int\_pnr=200;

class dom\_booking{

//Declaring variables

protected:

int d\_pnr,dom\_passengers,dom\_meal\_choice;

char flight\_dom[50],dom\_meal[20], meal1[20];;

char date\_dom[10];

char dom\_dep[20],dom\_arr[20];

int choice\_dom,source,destination;

public:

//Function for generating domestic ticket pnr numbers

void generate\_dom\_pnr(){

dom\_pnr++;

d\_pnr=dom\_pnr;

};

// Function for getting domestic travel information from the user

int travel\_details\_dom()

{

cout << "\nPlease enter date of departure (DD/MM/YY):" << endl;

cin >> date\_dom;

system("CLS");

cout << "\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t"<<endl;

cout<< " SOURCE "<<endl;

cout << "\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t\t"<<endl;

cout << " \t\t1.KATHMANDU(1)" << endl ;

cout <<" \t\t2.POKHARA(2)" << endl;

cout <<" \t\t3.BIRATNAGAR(3)" << endl;

cout <<" \t\t4.JANAKPUR (4) " << endl;

cout <<"\t\t5.RAJBIRAJ (5)" <<endl;

cout << "\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

cout << "Enter your source:" << endl;

cin >> source;

cout << "\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t"<<endl;

cout<< " DESTINATION "<<endl;

cout << "\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\t"<<endl;

cout << "1.KATHMANDU(1)" << endl;

cout <<"2.POKHARA(2)" << endl;

cout <<"3.BIRATNAGAR(3)" << endl;

cout <<"4.JANAKPUR(4)" << endl;

cout<<"5.RAJBIRAJ(5)"<<endl;

cout << "\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

cout << "Enter your destination :" << endl;

cin >> destination;

if((source==1 && destination==2) || (source==2 && destination==1))

{

cout << "\t \t \tAvailable flights" << endl << endl;

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

cout << "\nAIRLINES\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "\nYETI AIRLINES(1)\t08:00\t\t11:05\t\tRs.5000\n";

cout << "\nBUDDHA AIR(2)\t14:00\t\t17:05\t\tRs.5500\n";

cout << "\nSHREE AIRLINES(3)\t19:00\t\t22:05\t\tRs.6000\n";

cout <<"\nSAURYA AIRLINES(4)\t06:30\t\t9:15\t\tRs.7500\n";

}

else if((source==1 && destination==3) || (source==3 && destination==1))

{

cout << "\t \t \tAvailable flights" << endl << endl;

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

cout << "\nAirlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "\nYETI AIRLINES (1)\t08:00\t\t11:05\t\tRs.5000\n";

cout <<"\nBUDDHA AIRLINES(2)\t14:00\t\t17:05\t\tRs.5500\n";

cout << "\nSHREE AIRLINES(3)\t19:00\t\t22:05\t\tRs.6000\n";

cout <<"\nSAURYA AIRLINES(4)\t06:30\t\t9:15\t\tRs.7500\n";

}

else if((source==1 && destination==4) || (source==4 && destination==1))

{

cout << "\t \t \tAvailable flights" << endl << endl;

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

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;;

cout << "\nYETI AIRLINES(1)\t08:00\t\t11:05\t\tRs.5000\n";

cout << "\nBUDDHA AIR(2)\t14:00\t\t17:05\t\tRs.5500\n";

cout << "\nSHREE AIRLINES(3)t18:00\t\t21:05\t\tRs.6000\n";

cout <<"\nSAURYA AIRLINES(4)\t06:30\t\t9:15\t\tRs.7500\n";

}

else if((source==2 && destination==3) || (source==3 && destination==2))

{

cout << "\t \t \tAvailable flights" << endl << endl;

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

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "\nYETI AIRLINES(1)\t08:00\t\t11:05\t\tRs.5000\n";

cout << "\nBUDDHA AIR(2)\t14:00\t\t17:05\t\tRs.5500\n";

cout << "\nSHREE AIRLINES(3)\t18:00\t\t21:05\t\tRs.6000\n";

cout <<"\nSAURYA AIRLINES(4)\t06:30\t\t9:15\t\tRs.7500\n";

}

else if((source==2 && destination==4) || (source==4 && destination==2))

{

cout << "\t \t \tAvailable flights" << endl << endl;

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

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "\n1.YETI AIRLINES(1)\t08:00\t\t11:05\t\tRs.5000\n";

cout << "\nBUDDHA AIRLINES(2)\t14:00\t\t17:05\t\tRs.5500\n";

cout << "\nSHREE AIRLINES(3\t18:00\t\t21:05\t\tRs.6000\n";

cout <<"\nSAURYA AIRLINES(4)\t06:30\t\t9:15\t\tRs.7500\n";

}

else if((source==3 && destination==4) || (source==4 && destination==3))

{

cout << "\t \t \tAvailable flights" << endl << endl;

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

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "\nYETI AIRLINES(1)\t08:00\t\t11:05\t\tRs.5000\n";

cout << "\nBUDDHA AIRLINES(2)\t14:00\t\t17:05\t\tRs.5508\n";

cout << "\nSHREE AIRLINES(3)\t18:00\t\t21:05\t\tRs.6000\n";

cout <<"\nSAURYA AIRLINES(4)\t06:30\t\t9:15\t\tRs.7500\n";

}

else if(source==destination)

{

cout << "\nSource and destination are the same.Please enter try again.\n\n\n" << endl;

return travel\_details\_dom();

}

else

{

cout <<"\nYou have entered the wrong input entered.Please try again\n\n\n" << endl;

return travel\_details\_dom();

}

}

//Function for selecting domestic flight

int select\_flight\_dom()

{

cout << "\nEnter your choice : " << endl;

cin >> choice\_dom;

switch(choice\_dom)

{

case 1:

cout << "\nYour travel information"<<endl;

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

cout << "Flight name : YETI AIRLINES "<<endl;

strcpy(flight\_dom,"YETI AIRLINES");

cout << "Departure Time : 08:00"<<endl;

cout<<"Arrival Time: 11:05"<<endl;

strcpy(dom\_dep,"08:00");

strcpy(dom\_arr,"11:05");

break;

case 2:

cout << "\nYour travel information:"<<endl;

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

cout << "Flight name : BUDDHA AIR"<<endl;

strcpy(flight\_dom,"BUDDHA AIR");

cout << "Departure Time : 14:00"<<endl;

cout<<"Arrival Time: 17:05"<<endl;

strcpy(dom\_dep,"14:00");

strcpy(dom\_arr,"17:05");

break;

case 3:

cout << "\nYour travel information" << endl;

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

cout << "Flight name : SHREE AIRLINES" << endl;

strcpy(flight\_dom,"SHREE AIRLINES");

cout << "Departure Time : 18:00" << endl;

cout<<"Arrival Time: 21:05" << endl;

strcpy(dom\_dep,"18:00");

strcpy(dom\_arr,"21:05");

break;

case 4:

cout << "\nYour travel information" << endl;

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

cout << "Flight name : SAURYA AIRLINES" << endl;

strcpy(flight\_dom,"SAURYA AIRLINES");

cout << "Departure Time : 06:30" << endl;

cout<<"Arrival Time: 09:15" << endl;

strcpy(dom\_dep,"06:30");

strcpy(dom\_arr,"09:15");

break;

default:

cout << "\nYou have entered the wrong input entered.Try again\n\n\n" << endl;

return select\_flight\_dom();

}

}

int dom\_meals(){

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

cout <<"Please enter your meal preference : "<<endl;

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

cout << "1.Vegetarian meal(1): Rs.350 \n";

cout << "2.Non-Vegetarian meal(2) : Rs.450\n";

cout << "3.No meal(3)\n\n";

cout <<"Enter your choice : ";

cin >> dom\_meal\_choice;

while(dom\_meal\_choice>3 || dom\_meal\_choice<1){

cout<<"You have entered the wrong input.Please a number between 1 and 3\n";

cin>>dom\_meal\_choice;

}

{

if (dom\_meal\_choice == 1){

strcpy(meal1,"vegetarian meal");

}

else if (dom\_meal\_choice == 2){

strcpy(meal1,"non-Vegetarian meal");

}

else{

strcpy(meal1,"no meal");

}

cout<<"You have chosen "<<meal1<<endl;

}

}

};

//Class for international booking

class int\_booking

{

protected:

int i\_pnr,int\_passengers,int\_meal\_choice,int\_meal\_price;

char flight\_int[20],int\_dep[20],int\_arr[20],meal2[30],int\_meal\_choices[20];

char date\_int[20];

int source\_int,destination\_int,choice\_int;

public:

void generate\_int\_pnr()

{

int\_pnr++;

i\_pnr=int\_pnr;

}

// Function for getting international travel information from the user

int travel\_details\_int()

{

cout << "\nEnter date of departure (DD/MM/YY) :" << endl;

cin>>date\_int;

system("CLS");

system("CLS");

cout<< "\n SOURCE "<<endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout << "1.NEW-YORK(USA) (1)" << endl ;

cout << "2.CANBERRA(AUS) (2)" << endl ;

cout <<"3.Hong Kong(3)" << endl ;

cout <<"4.LONDON(UK)(4) " << endl ;

cout << "############################\n"<<endl;

cout << "Enter your source:" << endl;

cin >> source\_int;

cout<< "\n DESTINATION "<<endl;

cout << "#############################\n"<<endl;

cout << "1.NEW YORK(USA)(1)" << endl ;

cout <<"2. CANBERRA(AUS)(2)" << endl;

cout <<"3.Hong Kong(3)" << endl;

cout <<"4.LONDON(UK)(4) " << endl ;

cout << "##############################\n"<<endl;

cout << "Enter your destination : \n" << endl;

cin >> destination\_int;

if((source\_int==1 && destination\_int==2) || (source\_int==2 && destination\_int==1))

{

cout << "\t \t \tAvailable flights" << endl;

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

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "AMERICAN AIRLINES(1)\t04:10\t\t08:30\t\tRs.14,010\n";

cout << "EMIRATES AIRWAYS(2)\t14:00\t\t18:05\t\tRs.15,120\n";

cout << "QATAR AIRWAYS(3)\t22:00\t\t02:10\t\tRs.12,500\n";

cout << "BRITISH AIRWAYS(4) AIRWAYS(3)\t08:15\t\t12:30\t\tRs.17,500\n";

}

else if((source\_int==1 && destination\_int==3) || (source\_int==3 && destination\_int==1))

{

cout << "\t \t \tAvailable flights" << endl;

cout<<"------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*----------"<<endl;

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "AMERICAN AIRLINES(1)\t04:10\t\t08:30\t\tRs.23,010\n";

cout << "Emirates AIRWAYS(2)\t14:00\t\t18:05\t\tRs.28.340\n";

cout << "QATAR AIRWAYS(3)\t22:00\t\t02:10\t\tRs.29,320\n";

cout << "BRITISH AIRWAYS(4) AIRWAYS(3)\t08:15\t\t12:30\t\tRs.37,530\n";

}

else if((source\_int==1 && destination\_int==4) || (source\_int==4 && destination\_int==1))

{

cout << "\t \t \tAvailable flights" << endl;

cout<<"-------------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*--------------"<<endl;

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "AMERICAN AIRLINES(1)\t04:10\t\t08:30\t\tRs.40,010\n";

cout << "Emirates AIRWAYS(2)\t14:00\t\t18:05\t\tRs.50.340\n";

cout << "QATAR AIRWAYS(3)\t22:00\t\t02:10\t\tRs.35,320\n";

cout << "BRITISH AIRWAYS(4) AIRWAYS(3)\t08:15\t\t12:30\t\tRs.54,530\n";

}

else if((source\_int==2 && destination\_int==3) || (source\_int==3 && destination\_int==2))

{

cout << "\t \t \tAvailable flights" << endl;

cout<<"-----------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*-------------"<<endl;

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "AMERICAN AIRLINES(1)\t04:10\t\t08:30\t\tRs.23,010\n";

cout << "Emirates AIRWAYS(2)\t14:00\t\t18:05\t\tRs.28.340\n";

cout << "QATAR AIRWAYS(3)\t22:00\t\t02:10\t\tRs.29,320\n";

cout << "BRITISH AIRWAYS(4) AIRWAYS(3)\t08:15\t\t12:30\t\tRs.37,530\n";

}

else if((source\_int==2 && destination\_int==4) || (source\_int==4 && destination\_int==2))

{

cout << "\t \t \tAvailable flights" << endl;

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

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "AMERICAN AIRLINES(1)\t04:10\t\t08:30\t\tRs.23,010\n";

cout << "Emirates AIRWAYS(2)\t14:00\t\t18:05\t\tRs.28.340\n";

cout << "QATAR AIRWAYS(3)\t22:00\t\t02:10\t\tRs.29,320\n";

cout << "BRITISH AIRWAYS(4) AIRWAYS(3)\t08:15\t\t12:30\t\tRs.37,530\n";

}

else if((source\_int==3 && destination\_int==4) || (source\_int==4 && destination\_int==3))

{

cout << "\t \t \tAvailable flights" << endl;

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

cout << "Airlines\tDeparture\tArrival\t\tPrice\n"<<endl;

cout << "AMERICAN AIRLINES(1)\t04:10\t\t08:30\t\tRs.23,010\n";

cout << "Emirates AIRWAYS(2)\t14:00\t\t18:05\t\tRs.28.340\n";

cout << "QATAR AIRWAYS(3)\t22:00\t\t02:10\t\tRs.29,320\n";

cout << "BRITISH AIRWAYS(4) AIRWAYS(3)\t08:15\t\t12:30\t\tRs.37,530\n";

}

else if(source\_int==destination\_int)

{

cout << "\nSource and destination are the same.Please enter try again.\n\n\n" << endl;

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

return travel\_details\_int();

}

else

{

cout <<"\nYou have entered the wrong input entered.Try again\n\n\n" << endl;

return travel\_details\_int();

}

}

//Function for selecting international flight

int select\_flight\_int()

{

cout << "\nEnter your choice : " << endl;

cin >> choice\_int;

switch(choice\_int){

case 1:

cout << "\nYour travel information" <<endl;

cout<<"-----------------------------\n"<<endl;

cout << "Flight name :AMERICAN AIRLINES" << endl;

strcpy(flight\_int,"AMERICAN AIRLINES");

cout << "Departure Time: 04:10" << endl;

cout << "Arrival Time: 14:05" << endl;

strcpy(int\_dep,"10:00");

strcpy(int\_arr,"14:05");

break;

case 2:

cout << "\nYour travel information" << endl;

cout<<"-----------------------------\n"<<endl;

cout << "Flight name:Emirates" << endl;

strcpy(flight\_int,"Emirates");

cout << "Departure Time: 14:00" << endl;

cout << "Arrival Time: 18:05" << endl;

strcpy(int\_dep,"14:00");

strcpy(int\_arr,"18:05");

break;

case 3:

cout << "\nYour travel information" << endl;

cout<<"-----------------------------\n"<<endl;

cout << "Flight name:QATAR AIRWAYS" << endl;

strcpy(flight\_int,"QATAR AIRWAYS");

cout << "Departure Time : 18:00" << endl;

cout << "Arrival Time: 22:05" << endl;

strcpy(int\_dep,"18:00");

strcpy(int\_arr,"22:05");

break;

case 4:

cout << "\nYour travel information" << endl;

cout<<"-----------------------------\n"<<endl;

cout << "Flight name:BRITISH AIRWAYS" << endl;

strcpy(flight\_int,"BRITISH AIRWAYS");

cout << "Departure Time : 08:15" << endl;

cout << "Arrival Time: 12:30" << endl;

strcpy(int\_dep,"08:15");

strcpy(int\_arr,"12:30");

break;

default:

cout << "\nYou have entered the wrong input entered.Try again\n\n\n" << endl;

return select\_flight\_int();}

}

//Function for selecting international flight meal

int int\_meal(){

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

cout <<"Meal preferences "<<endl;

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

cout << "1.Vegetarian meal(1): Rs.450 \n";

cout << "2.Non-Vegetarian meal(2) : Rs.500\n";

cout << "3.No meal(3)\n\n";

cout <<"Enter your choice : ";

cin >> int\_meal\_choice;

while(int\_meal\_choice>3 ||int\_meal\_choice<1){

cout<<"You have entered the wrong input.Please enter a number between 1 and 3\n";

cin>>int\_meal\_choice;

}

{

if (int\_meal\_choice== 1){

strcpy(meal2,"vegetarian meal");

}

else if (int\_meal\_choice == 2){

strcpy(meal2,"non-vegetarian meal");

}

else{

strcpy(meal2,"no meal");

}

}

cout<<"You have chosen "<<meal2<<endl;

}

};

//Class passenger inherits the classes dom\_booking and int\_booking

class passenger: public dom\_booking,public int\_booking

{

protected:

char name[100],email[50],gender[20],pass\_no[30],contact\_no[30];

public:

void passenger\_details(int x)

{

system("CLS");

if(x==1)

{ travel\_details\_dom();

select\_flight\_dom();

dom\_meals();

}

else

{ travel\_details\_int();

select\_flight\_int();

int\_meal();

}

system("CLS");

cout << "\n\n\nEnter the required details:\n";

cout<<"\n\t|t------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*---\n"<<endl;

cout << "Enter your name:";

cin >> name;

cout << "Enter your email ID :";

cin >> email;

cout << "Enter your gender:";

cin >> gender;

cout<<"Enter your contact number:";

cin>>contact\_no;

cout<<"Enter your passport number:";

cin>>pass\_no;

}

void display\_details(){

system("CLS");

cout << "\n\nPassenger details\n";

cout<<"\n\t\t-------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*----\n"<<endl;;

cout << "Name:" <<name<< endl;

cout << "Gender:" << gender << endl;

cout << "Email ID:" << email << endl;

cout << "Contact No.:" << contact\_no << endl;

cout<<"Passport number :"<<pass\_no<<endl;

}

int getpnr\_dom()

{

return d\_pnr;

}

int getpnr\_int()

{

return i\_pnr;

}

void disp\_dom()

{

cout<<"PNR :" << d\_pnr << endl;

cout<<"Flight:" << flight\_dom << endl;

cout<<"Name:" <<name << endl;

cout<<"Boarding date:" << date\_dom << endl;

cout<<"Departure Time:" << dom\_dep << endl;

cout<<"Arrival Time:" << dom\_arr<<endl;

cout<<"Meal choice:"<<meal1<<endl;

}

//Function to display details of international booking

void disp\_int()

{

cout<<"PNR:" << i\_pnr << endl;

cout<<"Flight:" << flight\_int << endl;

cout<<"Name:" << name<<endl;

cout<<"Boarding date:" << date\_int << endl;

cout<<"Departure Time:" << int\_dep << endl;

cout<<"Arrival Time:" << int\_arr<<endl;

cout<<"Meal choice:"<<meal2<<endl;

}

};

//class for payment

class payment{

protected:

int choice3,bank,bank1,card,date,cvv,user\_id;

char password[30];

public:

void payment\_details()

{

system("CLS");

cout << "\n\n\nMode of payment\n";

cout << "\n1.Debit Card(1) \n2.Credit Card(2) \n3.Net Banking(3)";

cout << "\n\nEnter your choice :";

cin >> choice3;

switch(choice3)

{

case 1:

cout << "Enter card number:";

cin >> card;

cout << "Enter expiry date:";

cin >> date;

cout << "Enter CVV number:";

cin >> cvv;

cout << "Transaction Successful\n";

break;

case 2:

cout << "Enter card number:";

cin >> card;

cout << "Enter expiry date:";

cin >> date;

cout << "Transaction Successful\n";

break;

case 3:

cout << "\nBanks Available:\n-----\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*---- \n1.State bank of India(1) \n2ICICI(2) \n3.Axis Bank(3) \n4.HDFC(4)";

cout << "\nSelect your bank:";

cin >> bank;

cout << "\nYou have selected:" << bank;

cout << "\nEnter user id:";

cin >> user\_id;

cout << "Enter password:";

cin >> password;

cout<<"---\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*----";

cout << "\nTransaction Successful\n";

cout<<"-------\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*------";

break;

default:

cout << "\nWrong input entered.\nTry again\n";

return payment\_details();

}

}

};

void createfile\_dom(passenger y)

{ ofstream file\_out("Domestic\_flight\_details.txt",ios::binary|ios::app);

file\_out.write((char\*)&y,sizeof(y));

file\_out.close();

}

void dom\_cancelticket(int x)

{ passenger y;

int f=0;

ifstream file\_in("Domestic\_flight\_details.txt",ios::binary|ios::app);

ofstream file\_out("Domestic\_flight\_details1.txt",ios::binary|ios::app);

file\_in.read((char \*)&y,sizeof(y));

while(file\_in)

{

if(y.getpnr\_dom()!=x)

file\_out.write((char \*)&y,sizeof(y));

else

{

y.disp\_dom();

cout<<"\nYour ticket has been deleted\n";

f++;

}

file\_in.read((char \*)&y,sizeof(y));

}

if(f==0)

cout<<"Ticket not found\n";

file\_out.close();

file\_in.close();

remove("Domestic\_flight\_details.txt");

rename("Domestic\_flight\_details1.txt","Domestic\_flight\_details.txt");

}

void dom\_checkticket(int x)

{ passenger y;

int f=0;

ifstream file\_in("Domestic\_flight\_details.txt",ios::binary);

file\_in.read((char \*)&y,sizeof(y));

while(file\_in)

{

if(y.getpnr\_dom()==x)

{cout<<"\nTicket details\n"<<endl;

y.disp\_dom();

f++;

break;

}

file\_in.read((char \*)&y,sizeof(y));

}

file\_in.close();

if(f==0)

cout<<"This PNR number does not exist"<<endl;

}

void createfile\_int(passenger y)

{ ofstream file\_out("International.txt",ios::binary|ios::app);

file\_out.write((char\*)&y,sizeof(y));

file\_out.close();

}

void int\_cancelticket(int x)

{ passenger y;

int f=0;

ifstream file\_in("International.txt",ios::binary|ios::app);

ofstream file\_out("International1.txt",ios::binary|ios::app);

file\_in.read((char \*)&y,sizeof(y));

while(file\_in)

{

if(y.getpnr\_int()!=x)

file\_out.write((char \*)&y,sizeof(y));

else

{

y.disp\_int();

cout<<"You ticket is being deleted\n";

f++;

}

file\_in.read((char \*)&y,sizeof(y));

}

if(f==0)

cout<<"\nTicket not found\n";

file\_in.close();

file\_out.close();

remove("International.txt");

rename("International1.txt","International.txt");

}

void int\_checkticket(int x)

{ passenger y;

int f=0;

ifstream file\_in("International.txt",ios::binary);

file\_in.read((char \*)&y,sizeof(y));

while(file\_in)

{

if(y.getpnr\_int()==x)

{cout<<"\nTicket details\n"<<endl;

y.disp\_int();

f++;

break;

}

file\_in.read((char \*)&y,sizeof(y));

}

file\_in.close();

if(f==0)

cout<<"This PNR number does not exist"<<endl;

}

int main(){

class dom\_booking d;

class int\_booking i;

class passenger p;

class payment pay;

int choice,choice1,choice2,choice3,input;

char input1;

do{

system("CLS");

cout<<"\n\t\t------AIRLINE RESERVATION SYSTEM------\t\t"<<endl;

cout<<"\n\t\t----\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*---\t\t"<<endl;

cout << "1.Book flight(1) \n2.Cancel fight(2) \n3.Check your flight ticket(3) \n4.Exit(4)" << endl;

cout<<"-----\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*----"<<endl;

cout<<"Enter your choice:"<<endl;

cin>>choice;

switch(choice)

{

case 1:

system("CLS");

cout<<"Please enter your choice"<<endl;

cout << "1.Domestic Fights(1)\n 2.International Flights(2)" << endl;

cout << "\nEnter your option" << endl;

cin >> choice1;

switch(choice1){

//Booking domestic flight

case 1:

p.generate\_dom\_pnr();

p.passenger\_details(1);

pay.payment\_details();

p.display\_details();

p.disp\_dom();

createfile\_dom(p);

break;

case 2:

p.generate\_int\_pnr();

p.passenger\_details(2);

pay.payment\_details();

p.display\_details();

p.disp\_int();

createfile\_int(p);

break;

default:

cout << "Wrong input.\n\nPlease enter 1 for domestic flight booking and 2 for international flight booking" << endl;

return main();

}

break;

case 2:

system("CLS");

cout << "1.Domestic Fights(1) \n2.International Flights(2)" << endl;

cout << "\nPlease enter your option" << endl;

cin >> choice2;

if(choice2==1)

{

cout << "Enter your PNR number :" << endl;

cin>>input;

dom\_cancelticket(input);

}

else if(choice2==2)

{ cout << "Enter your PNR number :" << endl;

cin>>input;

int\_cancelticket(input);

}

else

{

cout << "Wrong input.\n";

return main();

}

break;

case 3:

system("CLS");

cout << "\nCheck your tickets" << endl;

cout<<"---\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*----"<<endl;

cout << "1.Domestic Fights(1) \n 2.International Flights(2)" << endl;

cout << "\nEnter your option :" << endl;

cin >> choice3;

if(choice3==1)

{cout << "\nEnter your PNR number :" << endl;

cin>>input;

dom\_checkticket(input);}

else if(choice3==2)

{ cout << "\nEnter your PNR number :" << endl;

cin>>input;

int\_checkticket(input);

}

else

{

cout << "Wrong input.";

return main();

}

break;

case 4:

system("CLS");

return 0;

default:

cout<<endl;

return main();

}

cout<<"\n\n\nDo you wish to continue? (Y/N)" << endl;

cin >> input1;

}while(input1=='Y' || input1=='y');

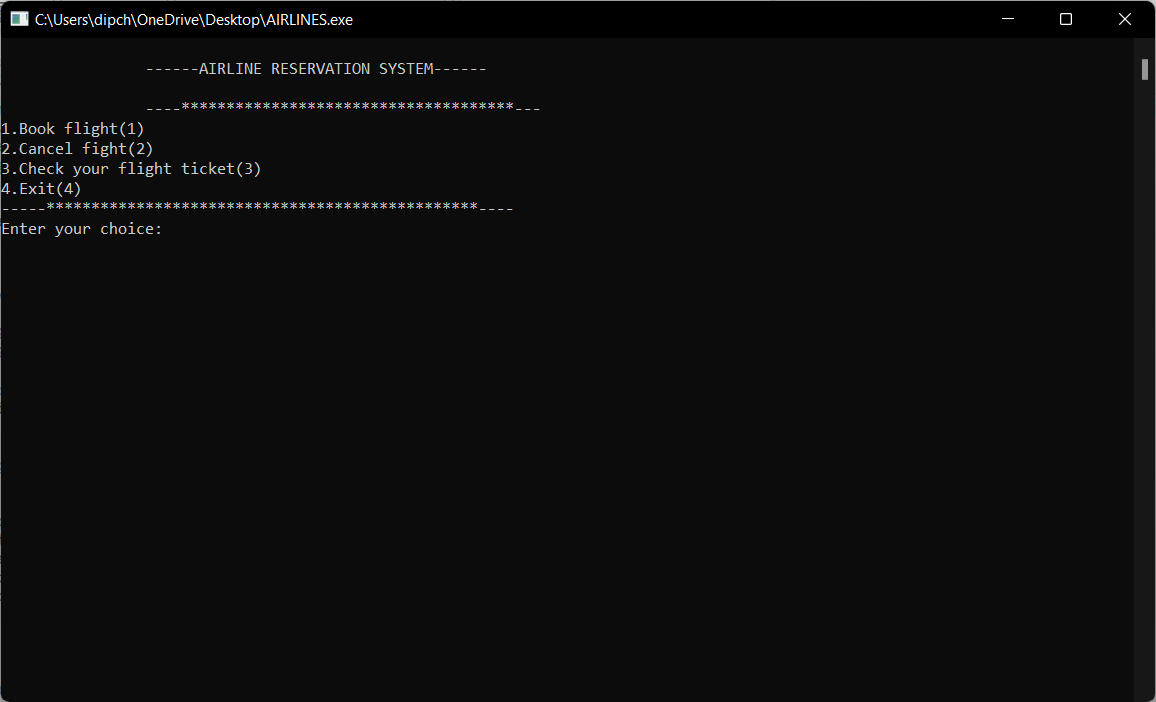
return 0;

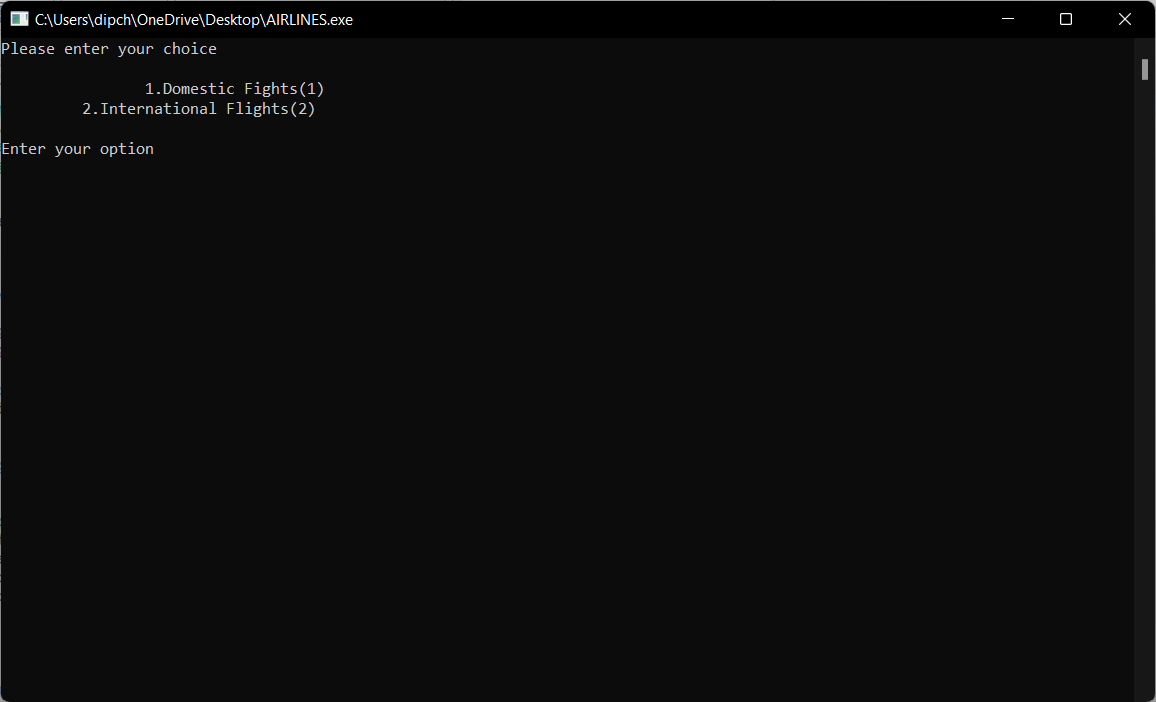
//getch();

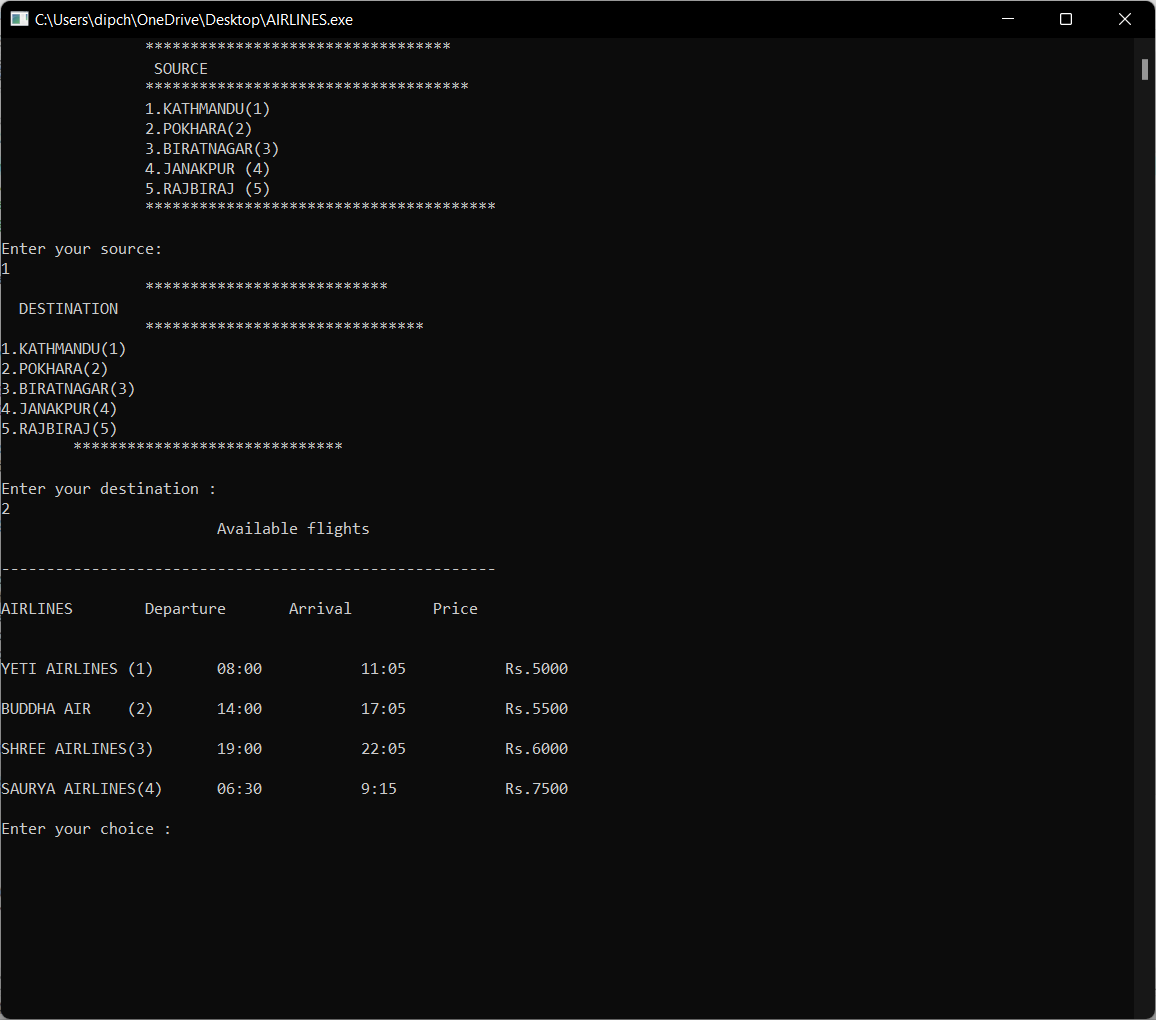
}

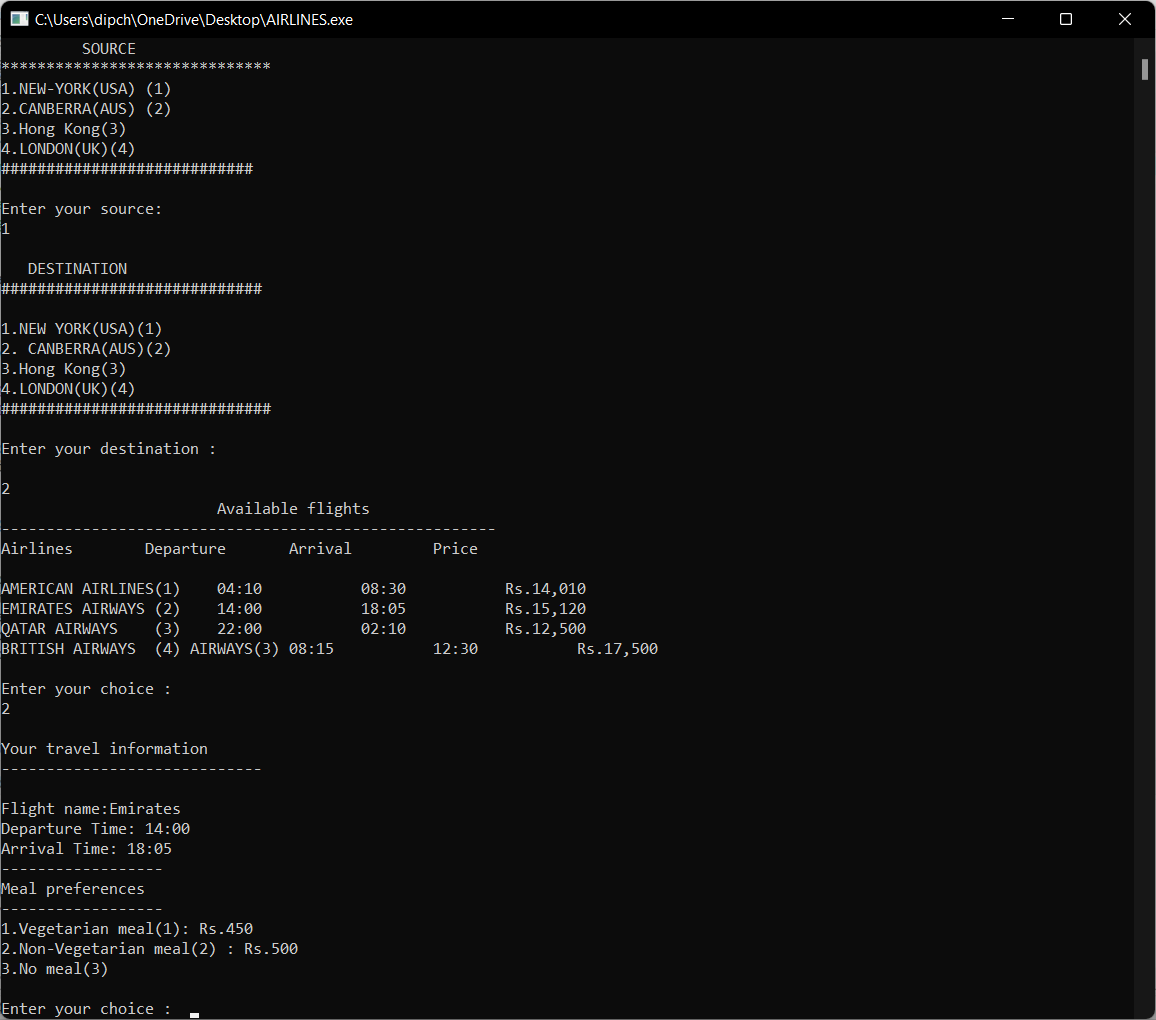
**TESTING AND OUTPUT**

Testing in a project development is very important task to find out the possible mistakes made by the developers . The project contain no error at all. This project has checked and the screenshots are given below:-









**References**

“Database System Concepts” by Abraham Silberschatz, Henry F. Korth, S. Sudharshan. “An Introduction to Database Systems”by C.J.Date, A.Kannan, S.Swamynathan. “Fundamentals of Database Systems” by Ramez Elmasri, Shamkant B. Navathe. “Database Management Systems” by Raghu Ramakrishnan.

“Database Systems Concepts, Design and Applications” by S.K. Singh