

```
//Ayesha Khalid
```

```
//ITSE
```

```
//Final project- airline reservation
```

```
#include <iostream>
```

```
#include <iomanip>
```

```
#include <string>
```

```
#include <string.h>
```

```
#include <cctype>
```

```
using namespace std;
```

```
//Given Function Prototypes
```

```
void displayWelcomeMessage();
```

```
void displayInputCommand(char&);
```

```
void displayCommands();
```

```
void displayFlightSchedule();
```

```
void displayErrorMessage();
```

```
void displayEndMessage();
```

```
int getFlightIndex();
```

```
void displaySeatingChart(int);
```

```
void displayPassengerList(int);
```

```
void bookReservation(int);
```

```
void cancelReservation(int);
```

```
int flightNumbers[8] = { 3548, 3488, 3498, 3644, 3487, 3497, 3645, 3549 };
```

```
char seats[4] = { 'A', 'B', 'C', 'D' };
```

```
int seatingChart[8][16][4] = { 0 };
```

```
string lastName[8][16][4] = {};
```

```
string firstName[8][16][4] = {};
```

```
int main()
```

```
{
```

```
    //declare variables
```

```
    int FlightIndex = 0;
```

```
    char command = ' ';
```

```
    for (int i = 0; i < 8; i++)
```

```
    {
```

```
        for (int j = 0; j < 16; j++)
```

```
        {
```

```
            for (int k = 0; k < 4; k++)
```

```
            {
```

```
                seatingChart[i][j][k] = 0;
```

```
            }
```

```
        }
```

```
    }
```

```
    //test data
```

```
    seatingChart[1][1][1] = 1;
```

```
    lastName[1][1][1] = "Khalid";
```

```
    firstName[1][1][1] = "Ayesha";
```

```
seatingChart[1][1][2] = 1;
lastName[1][1][2] = "Jordan";
firstName[1][1][2] = "Michael";
seatingChart[1][1][3] = 1;
lastName[1][1][3] = "Bird";
firstName[1][1][3] = "Larry";
seatingChart[1][15][1] = 1;
lastName[1][15][1] = "Johnson";
firstName[1][15][1] = "Magic";
```

```
displayWelcomeMessage();
system("cls");
displayCommands();
```

```
do
{
    displayInputCommand(command);

    switch (command)
    {
        case 'S':

            displayFlightSchedule();//Display FLight Schedule
            break;
        case 'D':

            displayFlightSchedule();
            FlightIndex = getFlightIndex();
            displaySeatingChart(FlightIndex);
```

```
        break;
case 'B':

    displayFlightSchedule();
    FlightIndex = getFlightIndex();
    displaySeatingChart(FlightIndex); //cancel reservation
    bookReservation(FlightIndex);
    break;
case 'C':

    displayFlightSchedule();
    FlightIndex = getFlightIndex();
    displaySeatingChart(FlightIndex);
    cancelReservation(FlightIndex);
    break;
case 'V':

    displayFlightSchedule();
    FlightIndex = getFlightIndex();// display passenger list
    displayPassengerList(FlightIndex);
    break;
case 'L':

    displayCommands(); //menu commands
    break;
case 'Q':

    displayEndMessage();//exit program
    break;
```

default:

```
    displayErrorMessage(); //incorrect command
```

```
    break;
```

```
}
```

```
} while (command != 'Q');
```

```
displayEndMessage();
```

```
system("pause");
```

```
return 0;
```

```
}
```

```
int getFlightIndex()
```

```
{
```

```
    int flightNum; //value of this variable is compared with values in flightNumbers array
```

```
    cout << "Please enter a flight number" << endl;
```

```
    cin >> flightNum; //input by user
```

```
    for (int i = 0; i < 8; i++) //this for loop searches through all 8 indexes of flightNumbers Array
```

```
    {
```

```
        if (flightNum == flightNumbers[i])
```

```
        {
```

```
            system("cls");
```

```
            cout << "Flight number found in index: " << i << endl;
```

```
            return i; // this returns index to main
```

```
    }

}

system("cls");

while (true)
{
    displayErrorMessage();
    cout << "Re-enter correct flight number" << endl;
    cin >> flightNum;

    for (int k = 0; k < 8; k++) //for loop to search the flightnumbers array
    {
        if (flightNum == flightNumbers[k])
        {

            cout << "Flight number found in index: " << k << endl;
            return k;

        }
    }

}

}
```

```

void displayPassengerList(int flightNumber)
{
    system("cls");
    int i=0; //counter variable

    cout << " _____" << endl << endl;

    cout << "\t" << "passenger list of flight number" << flightNumbers[flightNumber] << "\t" << endl;
    cout << " _____" << endl << endl;

    cout << "serial number" << "\t" << "passenger name" << "\t" << "\t" << "STATUS" << endl;
    cout << " _____" << endl << endl;

    while (i == flightNumber)
    {

        for (int seatNum = 1; seatNum < 16; seatNum++)

        {
            for (int seatchar = 0; seatchar < 4; seatchar++)
            {
                if (seatingChart[flightNumber][seatNum][seatchar] == 1) //condition for occupied seats
                {

```

```
        cout << "Seat :" << "(" << seatNum << seats[seatchar] << ")" "" \t" <<
firstName[flightNumber][seatNum][seatchar] << "\t" << lastName[flightNumber][seatNum][seatchar] <<
"\t" << "    OCCUPIED" << endl;
```

```
        cout << "_____ " << endl << endl;
```

```
    }
```

```
    else
```

```
    {
```

```
        cout << "Seat :" << seatNum << seats[seatchar] << "\t" << "    -    " << "\t" << "
UNOCCUPIED" << endl;
```

```
        cout << "_____ " << endl << endl;
```

```
    }
```

```
    }
```

```
    }
```

```
    i++;
```

```
    }
```

```
}
```

```
void displaySeatingChart(int flightNumber)
```

```
{
```



```
system("cls");
```

```
int i;
```

```
cout << " _____" << endl << endl;
```

```
cout << "\t" << "Seating chart of flight #" << flightNumbers[flightNumber] << "\t" << endl;
```

```
cout << " _____" << endl;
```

```
while (i == flightNumber)
```

```
{
```

```
    for (int seatNum = 1; seatNum < 17; seatNum++) // searches through seating chart array
```

```
    {
```

```
        for (int seatchar = 0; seatchar < 4; seatchar++)
```

```
        {
```

```
            if (seatingChart[flightNumber][seatNum][seatchar] == 1)
```

```
            {
```

```
                cout << "(" << seatNum << seats[seatchar] << ")" << "\t" << "\t";
```

```
            }
```

```
        else // what would happen when seats are unoccupied
```

```
        {
```

```
            cout << seatNum << seats[seatchar] << "\t" << "\t";
```

```

    }

}

    cout << endl << endl;
}
    i++;
}

}

```

```

void bookReservation(int flightNumber)
{
    int seatNumber = 0;
    char seatLetter;
    int i = 0;    //counter variable
    int seatsIndex;
    int sl = 0;
    cout << "enter your seat preference number between 1-16" << endl;
    cin >> seatNumber;

    if (seatNumber < 1 || seatNumber > 160)
    {
        do {

```

```

        cout << "Seat " << seatNumber << "seat doesn't exist" << endl << endl;

        cout << "re-enter seat number" << endl;

        cin >> seatNumber;

    } while (seatNumber < 0 || seatNumber > 16);

}

cout << "enter corresponding character of seat number from A-D " << endl;
cin >> seatLetter;

if (seatLetter > 100) //ascii code of 'D' is 100
{

    do {

        cout << "seat " << seatNumber << seatLetter << "doesn't exist" << endl;
        cout << "enter seat character again" << endl;
        cin >> seatLetter;

    } while (seatLetter > 100);

}

seatLetter = toupper(seatLetter);

for (sl = 0; sl < 4; sl++) // giving a value to seatletter which will be used as array subscript
{

```

```
    if (seatLetter == seats[sl])
    {
        seatsIndex = sl;
    }
}
```

```
cout << "enter your first name" << endl;
cin >> firstName[fliightNumber][seatNumber][seatsIndex];
```

```
cout << "enter your last name" << endl;
cin >> lastName[fliightNumber][seatNumber][seatsIndex];
```

```
seatingChart[fliightNumber][seatNumber][seatsIndex] == 1; //this seat is now occupied
```

```
cout << "dear " << firstName[fliightNumber][seatNumber][seatsIndex] << " " <<
lastName[fliightNumber][seatNumber][seatsIndex] << " "; //Displaying name of the User
```

```
cout << "your seat " << seatNumber << seatLetter << "has been confirmed"<<endl;
```

```
}
```

```
void cancelReservation(int flightNumber)
{
    int seatNumber = 0;
    char seatLetter;
```

```
int i = 0;    //counter variable
```

```
int seatsIndex;
```

```
int sl = 0;
```

```
cout << "enter your reserved seat number" << endl;
```

```
cin >> seatNumber;
```

```
if (seatNumber < 1 || seatNumber >16)
```

```
{
```

```
    do {
```

```
        cout << "seat " << seatNumber << "does not exist" << endl << endl;
```

```
        cout << "re-enter your seat number" << endl;
```

```
        cin >> seatNumber;
```

```
    } while (seatNumber < 0 || seatNumber >16); //keep taking input until condition is met
```

```
}
```

```
cout << "enter corresponding character of your reserved seat" << endl;
```

```
cin >> seatLetter;
```

```
if (seatLetter > 100)
```

```
{
```

```
    do {
```

```
        cout << "seat " << seatNumber << seatLetter << "seat doesn't exist" << endl;
```

```
        cout << "please enter youe seat character again" << endl;
```

```
        cin >> seatLetter;
```

```

        } while (seatLetter > 100); //condition

    }

    seatLetter = toupper(seatLetter);

    for (sl = 0; sl < 4; sl++)
    {
        if (seatLetter == seats[sl])
        {
            seatsIndex = sl;
        }
    }

    seatingChart[flightNumber][seatNumber][seatsIndex] == 0; //this seat is unoccupied

    cout << seatNumber << seatLetter << "your reservation has been cancelled" << endl;

}

```

```

void displayWelcomeMessage()
{
    cout << "*****" << endl;

    cout << "*" << endl;

    cout << "*" << "Welcome" << "*" << endl;
}

```

```

cout << "*"          to          "*" << endl;
cout << "*"      Tyler Eagle Jet      "*" << endl;
cout << "*"      Reservation System      "*" << endl;
cout << "*"          "*" << endl;
cout << "*"      8 Flights Daily      "*" << endl;
cout << "*"      between      "*" << endl;
cout << "*"      Tyler Pounds Field (TYR)      "*" << endl;
cout << "*"      Dallas/Fort Worth (DFW)      "*" << endl;
cout << "*"          "*" << endl;
cout << "*****" << endl;

system("pause");
}

void displayInputCommand(char& c)
{
    cout << endl << "Enter command (Press L to return to the menu) : ";

    cin >> c;
    c = toupper(c);
    system("cls");
}

void displayCommands()
{
    cout << "*****" << endl;
    cout << "*"          "*" << endl;
    cout << "*" (S)how flight schedules      "*" << endl;
    cout << "*" (B)ook a reservation      "*" << endl;

```

```

cout << "*" (C)ancel a reservation      "*" << endl;
cout << "*" (D)isplay seating chart      "*" << endl;
cout << "*" (V)iew passenger list        "*" << endl;
cout << "*" (L)ist commands               "*" << endl;
cout << "*" (Q)uit                       "*" << endl;
cout << "*"                             "*" << endl;
cout << "*****" << endl;
}

void displayFlightSchedule()
{
cout << "*****" << endl;
cout << "*" Tyler Pounds Field (TYR)      "*" << endl;
cout << "*"                             "*" << endl;
cout << "*" Flight no. Depart TYR/Arrive DFW  "*" << endl;
cout << "*"                             "*" << endl;
cout << "*" 3548 06:30am/07:15             "*" << endl;
cout << "*" 3488 10:55am/11:40             "*" << endl;
cout << "*" 3498 04:25pm/05:10             "*" << endl;
cout << "*" 3644 05:50pm/06:35             "*" << endl;
cout << "*"                             "*" << endl;
cout << "*"                             "*" << endl;
cout << "*" Dallas/Fort Worth (DFW)        "*" << endl;
cout << "*"                             "*" << endl;
cout << "*" Flight no. Depart DFW/Arrive TYR  "*" << endl;
cout << "*"                             "*" << endl;
cout << "*" 3487 09:45am/10:30am           "*" << endl;
cout << "*" 3497 03:15pm/04:00pm           "*" << endl;
cout << "*" 3645 04:40pm/05:25pm           "*" << endl;
cout << "*" 3549 08:10pm/08:55pm           "*" << endl;

```



```
cout << "*" << endl;
cout << "*" << endl;
cout << "*****" << endl;
}
```

```
void displayErrorMessage()
{
    cout << "\n==>Error: Command not recognized!" << endl;
    cout << "\n\n";
}
```

```
void displayEndMessage()
{
    system("cls");
    cout << endl << endl << endl << endl;
    cout << "Thanks for using our flight system";
    cout << endl << "come visit us again!!!" << endl;
    cout << "\n\n\n\n";
}
```