#include <iostream>

#include <string.h>

#include <conio.h>

#define max 100

using namespace std;

// Class Customer

class Customer

{

public:

    char name[100];

    char address[100];

    char phone[12];

    char from\_date[20];

    char to\_date[20];

    float payment\_advance;

    int booking\_id;

};

class Room

{

public:

    char type;

    char stype;

    char ac;

    int roomNumber;

    int rent;

    int status;

    class Customer cust;

    class Room addRoom(int);

    void searchRoom(int);

    void deleteRoom(int);

    void displayRoom(Room);

};

// Global Declarations

class Room rooms[max];

int count = 0;

Room Room::addRoom(int rno)

{

    class Room room;

    room.roomNumber = rno;

    cout << "\nType AC/Non-AC (A/N) : ";

    cin >> room.ac;

    cout << "\nType Comfort (S/N) : ";

    cin >> room.type;

    cout << "\nType Size (B/S) : ";

    cin >> room.stype;

    cout << "\nDaily Rent : ";

    cin >> room.rent;

    room.status = 0;

    cout << "\n Room Added Successfully!";

    getch();

    return room;

}

void Room::searchRoom(int rno)

{

    int i, found = 0;

    for (i = 0; i < count; i++)

    {

        if (rooms[i].roomNumber == rno)

        {

            found = 1;

            break;

        }

    }

    if (found == 1)

    {

        cout << "Room Details\n";

        if (rooms[i].status == 1)

        {

            cout << "\nRoom is Reserved";

        }

        else

        {

            cout << "\nRoom is available";

        }

        displayRoom(rooms[i]);

        getch();

    }

    else

    {

        cout << "\nRoom not found";

        getch();

    }

}

void Room::displayRoom(Room tempRoom)

{

    cout << "\nRoom Number: \t" << tempRoom.roomNumber;

    cout << "\nType AC/Non-AC (A/N) " << tempRoom.ac;

    cout << "\nType Comfort (S/N) " << tempRoom.type;

    cout << "\nType Size (B/S) " << tempRoom.stype;

    cout << "\nRent: " << tempRoom.rent;

}

// hotel management class

class HotelMgnt : protected Room

{

public:

    void checkIn();

    void getAvailRoom();

    void searchCustomer(char \*);

    void checkOut(int);

    void guestSummaryReport();

};

void HotelMgnt::guestSummaryReport()

{

    if (count == 0)

    {

        cout << "\n No Guest in Hotel !!";

    }

    for (int i = 0; i < count; i++)

    {

        if (rooms[i].status == 1)

        {

            cout << "\n Customer First Name : " << rooms[i].cust.name;

            cout << "\n Room Number : " << rooms[i].roomNumber;

            cout << "\n Address (only city) : " << rooms[i].cust.address;

            cout << "\n Phone : " << rooms[i].cust.phone;

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

        }

    }

    getch();

}

// hotel management reservation of room

void HotelMgnt::checkIn()

{

    int i, found = 0, rno;

    class Room room;

    cout << "\nEnter Room number : ";

    cin >> rno;

    for (i = 0; i < count; i++)

    {

        if (rooms[i].roomNumber == rno)

        {

            found = 1;

            break;

        }

    }

    if (found == 1)

    {

        if (rooms[i].status == 1)

        {

            cout << "\nRoom is already Booked";

            getch();

            return;

        }

        cout << "\nEnter booking id: ";

        cin >> rooms[i].cust.booking\_id;

        cout << "\nEnter Customer Name (First Name): ";

        cin >> rooms[i].cust.name;

        cout << "\nEnter Address (only city): ";

        cin >> rooms[i].cust.address;

        cout << "\nEnter Phone: ";

        cin >> rooms[i].cust.phone;

        cout << "\nEnter From Date: ";

        cin >> rooms[i].cust.from\_date;

        cout << "\nEnter to  Date: ";

        cin >> rooms[i].cust.to\_date;

        cout << "\nEnter Advance Payment: ";

        cin >> rooms[i].cust.payment\_advance;

        rooms[i].status = 1;

        cout << "\n Customer Checked-in Successfully..";

        getch();

    }

}

// hotel management shows available rooms

void HotelMgnt::getAvailRoom()

{

    int i, found = 0;

    for (i = 0; i < count; i++)

    {

        if (rooms[i].status == 0)

        {

            displayRoom(rooms[i]);

            cout << "\n\nPress enter for next room";

            found = 1;

            getch();

        }

    }

    if (found == 0)

    {

        cout << "\nAll rooms are reserved";

        getch();

    }

}

// hotel management shows all persons that have booked room

void HotelMgnt::searchCustomer(char \*pname)

{

    int i, found = 0;

    for (i = 0; i < count; i++)

    {

        if (rooms[i].status == 1 && stricmp(rooms[i].cust.name, pname) == 0)

        {

            cout << "\nCustomer Name: " << rooms[i].cust.name;

            cout << "\nRoom Number: " << rooms[i].roomNumber;

            cout << "\n\nPress enter for next record";

            found = 1;

            getch();

        }

    }

    if (found == 0)

    {

        cout << "\nPerson not found.";

        getch();

    }

}

// hotel managemt generates the bill of the expenses

void HotelMgnt::checkOut(int roomNum)

{

    int i, found = 0, days, rno;

    float billAmount = 0;

    for (i = 0; i < count; i++)

    {

        if (rooms[i].status == 1 && rooms[i].roomNumber == roomNum)

        {

            // rno = rooms[i].roomNumber;

            found = 1;

            // getch();

            break;

        }

    }

    if (found == 1)

    {

        cout << "\nEnter Number of Days:\t";

        cin >> days;

        billAmount = days \* rooms[i].rent;

        cout << "\n\t######## CheckOut Details ########\n";

        cout << "\nCustomer Name : " << rooms[i].cust.name;

        cout << "\nRoom Number : " << rooms[i].roomNumber;

        cout << "\nAddress : " << rooms[i].cust.address;

        cout << "\nPhone : " << rooms[i].cust.phone;

        cout << "\nTotal Amount Due : " << billAmount << " /";

        cout << "\nAdvance Paid: " << rooms[i].cust.payment\_advance << " /";

        cout << "\n\*\*\* Total Payable: " << billAmount - rooms[i].cust.payment\_advance << "/ only";

        rooms[i].status = 0;

    }

    getch();

}

// managing rooms (adding and searching available rooms)

void manageRooms()

{

    class Room room;

    int opt, rno, i, flag = 0;

    char ch;

    do

    {

        system("cls");

        cout << "\n### Manage Rooms ###";

        cout << "\n1. Add Room";

        cout << "\n2. Search Room";

        cout << "\n3. Back to Main Menu";

        cout << "\n\nEnter Option: ";

        cin >> opt;

        // switch statement

        switch (opt)

        {

        case 1:

            cout << "\nEnter Room Number: ";

            cin >> rno;

            i = 0;

            for (i = 0; i < count; i++)

            {

                if (rooms[i].roomNumber == rno)

                {

                    flag = 1;

                }

            }

            if (flag == 1)

            {

                cout << "\nRoom Number is Present.\nPlease enter unique Number";

                flag = 0;

                getch();

            }

            else

            {

                rooms[count] = room.addRoom(rno);

                count++;

            }

            break;

        case 2:

            cout << "\nEnter room number: ";

            cin >> rno;

            room.searchRoom(rno);

            break;

        case 3:

            // nothing to do

            break;

        default:

            cout << "\nPlease Enter correct option";

            break;

        }

    } while (opt != 3);

}

using namespace std;

int main()

{

    class HotelMgnt hm;

    int i, j, opt, rno;

    char ch;

    char pname[100];

    system("cls");

    do

    {

        system("cls");

        cout << "######## Hotel Management #########\n";

        cout << "\n1. Manage Rooms";

        cout << "\n2. Check-In Room";

        cout << "\n3. Available Rooms";

        cout << "\n4. Search Customer";

        cout << "\n5. Check-Out Room";

        cout << "\n6. Guest Summary Report";

        cout << "\n7. Exit";

        cout << "\n\nEnter Option: ";

        cin >> opt;

        switch (opt)

        {

        case 1:

            manageRooms();

            break;

        case 2:

            if (count == 0)

            {

                cout << "\nRooms data is not available.\nPlease add the rooms first.";

                getch();

            }

            else

                hm.checkIn();

            break;

        case 3:

            if (count == 0)

            {

                cout << "\nRooms data is not available.\nPlease add the rooms first.";

                getch();

            }

            else

                hm.getAvailRoom();

            break;

        case 4:

            if (count == 0)

            {

                cout << "\nRooms are not available.\nPlease add the rooms first.";

                getch();

            }

            else

            {

                cout << "Enter Customer Name: ";

                cin >> pname;

                hm.searchCustomer(pname);

            }

            break;

        case 5:

            if (count == 0)

            {

                cout << "\nRooms are not available.\nPlease add the rooms first.";

                getch();

            }

            else

            {

                cout << "Enter Room Number : ";

                cin >> rno;

                hm.checkOut(rno);

            }

            break;

        case 6:

            hm.guestSummaryReport();

            break;

        case 7:

            cout << "\nTHANK YOU! FOR USING SOFTWARE";

            break;

        default:

            cout << "\nPlease Enter correct option";

            break;

        }

    } while (opt != 7);

    getch();

}

