C++ Project
Hotel Management

```
#include<iostream>
#include<string.h>
#include<cctype>
#define max 100
using namespace std;
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 displayRoomdetails(Room);
};
class HotelMgnt : protected Room
{
public:
    void checkInroom();
    void getAvailRoom();
    void searchCustomer(char*);
    void checkOutroom(int);
    void guestSummaryReport();
};
int count = 0;
class Room rooms[max];
```

```
class Room Room::addRoom(int rno)
{
    class Room room;
    room.roomNumber = rno;
    cout << "\nplease enter the type of room (AC/N-AC) : ";
    cin >> room.ac;
    cout << "\nplease enter the luxury (Delux/Normal) : ";
    cin >> room.type;
    cout << "\nplease enter the size of the room (B/S) : ";
    cin >> room.stype;
    cout << "\n enter the Daily Rent of the room : ";
    cin >> room.rent;
    room.status = 0;
    cout << "\n Room Added Successfully!";
    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";
        }
        displayRoomdetails(rooms[i]);
    }
    else
    {
        cout << "\nRoom not found";
    }
}
void Room::displayRoomdetails(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;
}
void HotelMgnt::checkInroom()
{
    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 (rooms[i].roomNumber != rno)
     {
        cout << "Room with that number not available\nenter correct number\n";
    }
    if (found == 1)
    {
        if (rooms[i].status == 1)
         {
            cout << "\nRoom chosen by you is already Booked";
            return;
        }
        cout << "\nEnter booking id: ";
        cin >> rooms[i].cust.booking_id;
        bool validName = false;
        do
         {
            cout << "\nEnter Customer Name (alphabets only): ";
            cin.ignore();
            cin.getline(rooms[i].cust.name, sizeof(rooms[i].cust.name));
            validName = true;
            for (int j = 0; j < strlen(rooms[i].cust.name); j++)
            {
                if (!isalpha(rooms[i].cust.name[j]))
                {
                    validName = false;
                    break;
                }
            }
            if (!validName)
```

```
            {
                cout << "Invalid name! Please enter a name without numerical
characters.\n";
            }

        } while (!validName);
        cout << "\nEnter Address of the customer(only city): ";
        cin.getline(rooms[i].cust.address, sizeof(rooms[i].cust.address));
        cout << "\nEnter Phone number of the customer: ";
        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..";
    }
}
void HotelMgnt::getAvailRoom()
{
    int i, found = 0;
    for (i = 0; i < count; i++)
    {
        if (rooms[i].status == 0)
        {
            displayRoomdetails(rooms[i]);
            cout << "\n\nPress enter for next room";
            found = 1;
        }
    }
    if (found == 0)
    {
        cout << "\nAll rooms are reserved";
    }
}
void HotelMgnt::searchCustomer(char* pname)
{
    int i, found = 0;
    for (i = 0; i < count; i++)
    {
        if (rooms[i].status == 1 && strcmp(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\n";
            found = 1;
```

```
        }
    }
    if (found == 0)
    {
        cout << "\nPerson not found.\n";
    }
}
void HotelMgnt::checkOutroom(int roomNum)
{
    int i, found = 0, days;
    float billAmount = 0;
    for (i = 0; i < count; i++)
    {
        if (rooms[i].status == 1 && rooms[i].roomNumber == roomNum)
        {
            found = 1;
            break;
        }
    }
    if (found == 0)
     {
        cout << "Room with that number not available\n please enter correct
room number\n";
    }
    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;
    }
}
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 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;
        }
    }
}
void manageRooms()
{
    class Room room;
    int opt, rno, i, flag = 0;
    do
    {
        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 (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 already Present.\nPlease enter
unique Number";
                flag = 0;
            }
            else
            {
                rooms[count] = room.addRoom(rno);
                count++;
            }
            break;
        case 2:
```

```
            cout << "\nEnter room number you want to search: ";
            cin >> rno;
            room.searchRoom(rno);
            break;
        case 3:
            break;
        default:
            cout << "\nPlease Enter correct option";
            break;
        }
    } while (opt != 3);
}
int main()
{
    class HotelMgnt hm;
    int i, j, opt, rno;
    char pname[100];
    do
    {
        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.";
            }
            else
                hm.checkInroom();
            break;
        case 3:
            if (count == 0)
            {
                cout << "\nRooms data is not available.\nPlease add the rooms
first.";
            }
```

```
            else
                hm.getAvailRoom();
            break;
        case 4:
            if (count == 0)
            {
                cout << "\nRooms are not available.\nPlease add the rooms
first.";
            }
            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.";
            }
            else {
                cout << "Enter Room Number : ";
                cin >> rno;
                hm.checkOutroom(rno);
            }
            break;
        case 6:
            hm.guestSummaryReport();
            break;
            case 7:
            break;
        default:
            cout << "\nPlease Enter correct option";
            break;
        }
    } while (opt != 7);
    return 0;
}
```