**Lab Report**

**Course Tittle:** Object Oriented Programming Lab

**Course Code:** CSE 215

**Experiment No:** 10

**Experiment Name:** Designing Student Management System by utilizing Encapsulation, Dependency and Collection Framework of JAVA

**Submitted To:**

**Name:** Mst. Umme Ayman

**Designation:** Lecturer

**Department of CSE**

**Daffodil International University**

**Submitted By**

**Name:** Md Raduan Ahamed

**ID:** 0242220005101839

**Section:** 63\_O

**Department of CSE**

**Daffodil International University**

**Submission Date:** 29-11-2023

You are asked with designing a Hotel Management System for a hotel. This system should have classes to manage Room, Guest, Reservation.Where Room has information like room\_no, room\_type, rate, availability status.Guests can reserve the room.The system also keeps tracks of Reservation.

1.Design the UML diagram of above mentioned scenario

- roomNo: int

- roomType: String

- rate: double

- isAvailable: boolean

**Guest**

**Room**

- guestID: int

- guestName: String

←

**HotelManagementSystem**

+static void main(String[] args)

+ getGuestID(): int

+ getGuestName(): String

+ setGuestName(name: String)

+ getRoomNo(): int

+ getRoomType(): String

+ getRate(): double

+ isAvailable(): boolean

+ reserveRoom(): void



**Reservation**



- reservationID: int

- room: Room

- guest: Guest

- checkInDate: Date

- checkOutDate: Date



+ getReservationID(): int

+ getRoom(): Room

+ getGuest(): Guest

+ getCheckInDate(): Date

+ getCheckOutDate(): Date

2.Write the java code of above UML.

package lab\_11;  
  
import java.util.ArrayList;  
import java.util.List;  
  
class Room {  
 private int roomNo;  
 private String roomType;  
 private double rate;  
 private boolean isAvailable;  
  
 public Room(int roomNo, String roomType, double rate) {  
 this.roomNo = roomNo;  
 this.roomType = roomType;  
 this.rate = rate;  
 this.isAvailable = true; *// Room is initially available* }  
  
 public int getRoomNo() {  
 return roomNo;  
 }  
  
 public String getRoomType() {  
 return roomType;  
 }  
  
 public double getRate() {  
 return rate;  
 }  
  
 public boolean isAvailable() {  
 return isAvailable;  
 }  
  
 public void setAvailable(boolean available) {  
 isAvailable = available;  
 }  
}

package lab\_11;  
  
class Guest {  
 private String guestName;  
  
 public Guest(String guestName) {  
 this.guestName = guestName;  
 }  
  
 public String getGuestName() {  
 return guestName;  
 }  
}

package lab\_11;  
  
class Reservation {  
 private Guest guest;  
 private Room room;  
  
 public Reservation(Guest guest, Room room) {  
 this.guest = guest;  
 this.room = room;  
 }  
  
 public Guest getGuest() {  
 return guest;  
 }  
  
 public Room getRoom() {  
 return room;  
 }  
}

package lab\_11;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class hotel\_management{  
 private List<Room> rooms;  
 private List<Reservation> reservations;  
  
 public hotel\_management() {  
 rooms = new ArrayList<>();  
 reservations = new ArrayList<>();  
 }  
  
 public void addRoom(Room room) {  
 rooms.add(room);  
 }  
  
 public void displayAvailableRooms() {  
 System.*out*.println("Available Rooms:");  
 for (Room room : rooms) {  
 if (room.isAvailable()) {  
 System.*out*.println("Room No: " + room.getRoomNo() +  
 ", Type: " + room.getRoomType() +  
 ", Rate: $" + room.getRate());  
 }  
 }  
 }  
  
 public void makeReservation(Guest guest, Room room) {  
 if (room.isAvailable()) {  
 room.setAvailable(false);  
 Reservation reservation = new Reservation(guest, room);  
 reservations.add(reservation);  
 System.*out*.println("Reservation made for " + guest.getGuestName() +  
 " in Room No " + room.getRoomNo());  
 } else {  
 System.*out*.println("Room is not available for reservation.");  
 }  
 }  
  
 public static void main(String[] args) {  
 hotel\_management hotelSystem = new hotel\_management();  
  
 *// Adding rooms to the system* hotelSystem.addRoom(new Room(111, "Single", 300.0));  
 hotelSystem.addRoom(new Room(112, "Double", 750.0));  
 hotelSystem.addRoom(new Room(113, "Suite", 600.0));  
  
 *// Display available rooms* hotelSystem.displayAvailableRooms();  
  
 *// Making a reservation* Guest guest1 = new Guest("John Doe");  
 hotelSystem.makeReservation(guest1, hotelSystem.rooms.get(0));  
  
 *// Display available rooms after reservation* hotelSystem.displayAvailableRooms();  
 }  
}

**Output:**

Available Rooms:

Room No: 111, Type: Single, Rate: $300.0

Room No: 112, Type: Double, Rate: $750.0

Room No: 113, Type: Suite, Rate: $600.0

Reservation made for John Doe in Room No 111

Available Rooms:

Room No: 112, Type: Double, Rate: $750.0

Room No: 113, Type: Suite, Rate: $600.0