```
import java.util.ArrayList;
import java.util.Scanner;

class Room {
    private int roomNumber;
    private String roomType;
    private boolean isAvailable;
    private double price;

    public Room(int roomNumber, String roomType, double price) {
        this.roomNumber = roomNumber;
        this.roomType = roomType;
        this.price = price;
        this.isAvailable = true;
    }

    public int getRoomNumber() {
        return roomNumber;
    }

    public String getRoomType() {
        return roomType;
    }

    public boolean isAvailable() {
        return isAvailable;
    }

    public void bookRoom() {
        isAvailable = false;
    }
```

```
    public void vacateRoom() {
        isAvailable = true;
    }

    public double getPrice() {
        return price;
    }

    @Override
    public String toString() {
        return "Room " + roomNumber + " (" + roomType + ") - " + (isAvailable ? "Available" : "Booked");
    }
}

class Customer {
    private String name;
    private String contactNumber;
    private String address;
    private Room room;

    public Customer(String name, String contactNumber, String address) {
        this.name = name;
        this.contactNumber = contactNumber;
        this.address = address;
        this.room = null;
    }

    public String getName() {
        return name;
    }
```

```
    public String getContactNumber() {
        return contactNumber;
    }

    public String getAddress() {
        return address;
    }

    public void setRoom(Room room) {
        this.room = room;
    }

    public Room getRoom() {
        return room;
    }

    @Override
    public String toString() {
        return "Customer Name: " + name + ", Contact: " + contactNumber + ", Address: " + address +
                (room != null ? ", Room: " + room.getRoomNumber() : ", No Room Booked");
    }
}

public class HotelManagementSystem {
    private ArrayList<Room> rooms;
    private ArrayList<Customer> customers;

    public HotelManagementSystem() {
        rooms = new ArrayList<>();
        customers = new ArrayList<>();
```

```
}

public void addRoom(int roomNumber, String roomType, double price) {
    rooms.add(new Room(roomNumber, roomType, price));
}

public void addCustomer(String name, String contactNumber, String address) {
    customers.add(new Customer(name, contactNumber, address));
}

public void bookRoom(String customerName, int roomNumber) {
    Customer customer = findCustomerByName(customerName);
    Room room = findRoomByNumber(roomNumber);

    if (customer == null) {
        System.out.println("Customer not found.");
        return;
    }
    if (room == null || !room.isAvailable()) {
        System.out.println("Room is not available.");
        return;
    }

    customer.setRoom(room);
    room.bookRoom();
    System.out.println("Room " + roomNumber + " successfully booked for " + customerName);
}

public void vacateRoom(int roomNumber) {
    Room room = findRoomByNumber(roomNumber);
    if (room == null || room.isAvailable()) {
```

```
            System.out.println("Room is already vacant.");
            return;
        }

        Customer customer = findCustomerByRoom(room);
        if (customer != null) {
            customer.setRoom(null);
        }

        room.vacateRoom();
        System.out.println("Room " + roomNumber + " is now vacant.");
    }

    public void displayRooms() {
        for (Room room : rooms) {
            System.out.println(room);
        }
    }

    public void displayCustomers() {
        for (Customer customer : customers) {
            System.out.println(customer);
        }
    }

    private Room findRoomByNumber(int roomNumber) {
        for (Room room : rooms) {
            if (room.getRoomNumber() == roomNumber) {
                return room;
            }
        }
```

```
        return null;
    }

    private Customer findCustomerByName(String name) {
        for (Customer customer : customers) {
            if (customer.getName().equalsIgnoreCase(name)) {
                return customer;
            }
        }
        return null;
    }

    private Customer findCustomerByRoom(Room room) {
        for (Customer customer : customers) {
            if (customer.getRoom() == room) {
                return customer;
            }
        }
        return null;
    }

    public static void main(String[] args) {
        HotelManagementSystem system = new HotelManagementSystem();
        Scanner scanner = new Scanner(System.in);
        int choice;

        // Adding some sample rooms
        system.addRoom(101, "Single", 100.0);
        system.addRoom(102, "Double", 150.0);
        system.addRoom(103, "Suite", 200.0);
```

```
do {
    System.out.println("\n--- Hotel Management System ---");
    System.out.println("1. Display Rooms");
    System.out.println("2. Display Customers");
    System.out.println("3. Add Customer");
    System.out.println("4. Book Room");
    System.out.println("5. Vacate Room");
    System.out.println("6. Exit");
    System.out.print("Enter your choice: ");
    choice = scanner.nextInt();
    scanner.nextLine(); // Consume newline

    switch (choice) {
        case 1:
            system.displayRooms();
            break;
        case 2:
            system.displayCustomers();
            break;
        case 3:
            System.out.print("Enter Customer Name: ");
            String name = scanner.nextLine();
            System.out.print("Enter Contact Number: ");
            String contactNumber = scanner.nextLine();
            System.out.print("Enter Address: ");
            String address = scanner.nextLine();
            system.addCustomer(name, contactNumber, address);
            System.out.println("Customer added successfully.");
            break;
        case 4:
            System.out.print("Enter Customer Name: ");
```

```
                name = scanner.nextLine();
                System.out.print("Enter Room Number: ");
                int roomNumber = scanner.nextInt();
                system.bookRoom(name, roomNumber);
                break;
            case 5:
                System.out.print("Enter Room Number to Vacate: ");
                roomNumber = scanner.nextInt();
                system.vacateRoom(roomNumber);
                break;
            case 6:
                System.out.println("Exiting...");
                break;
            default:
                System.out.println("Invalid choice, please try again.");
        }
    } while (choice != 6);

    scanner.close();
  }
}
```