import java.util.\*;

class Room {

int number;

String type;

double pricePerNight;

boolean isBooked;

Room(int number, String type, double price) {

this.number = number;

this.type = type;

this.pricePerNight = price;

this.isBooked = false;

}

void display() {

System.out.printf("Room %d | Type: %s | ₹%.2f\n", number, type, pricePerNight);

}

}

class Booking {

String bookingId;

String guestName;

String phone;

int roomNumber;

int nights;

double totalAmount;

boolean isCheckedOut;

Booking(String bookingId, String guestName, String phone, int roomNumber, int nights, double totalAmount) {

this.bookingId = bookingId;

this.guestName = guestName;

this.phone = phone;

this.roomNumber = roomNumber;

this.nights = nights;

this.totalAmount = totalAmount;

this.isCheckedOut = false;

}

void display() {

System.out.println("Booking ID: " + bookingId);

System.out.println("Guest Name: " + guestName);

System.out.println("Phone: " + phone);

System.out.println("Room Number: " + roomNumber);

System.out.println("Nights: " + nights);

System.out.println("Total: ₹" + totalAmount);

System.out.println("Status: " + (isCheckedOut ? "Checked Out" : "Active"));

System.out.println("--------------------------------------");

}

}

class Review {

String guestName;

String comment;

Review(String guestName, String comment) {

this.guestName = guestName;

this.comment = comment;

}

void display() {

System.out.println("⭐ " + guestName + ": " + comment);

}

}

public class Main {

static String hotelName = "Grand Stay Hotel";

static List<Room> rooms = new ArrayList<>();

static List<Booking> bookings = new ArrayList<>();

static List<Review> reviews = new ArrayList<>();

static Scanner sc = new Scanner(System.in);

public static void main(String[] args) {

initializeRooms();

int choice;

do {

System.out.println("\n===== Welcome to " + hotelName + " =====");

System.out.println("1. View Available Rooms");

System.out.println("2. Book a Room");

System.out.println("3. Cancel Booking");

System.out.println("4. View All Bookings");

System.out.println("5. View Guest Reviews");

System.out.println("6. Checkout");

System.out.println("7. Exit");

System.out.print("Enter your choice: ");

choice = sc.nextInt();

sc.nextLine();

switch (choice) {

case 1:

viewAvailableRooms();

break;

case 2:

bookRoom();

break;

case 3:

cancelBooking();

break;

case 4:

viewBookings();

break;

case 5:

viewReviews();

break;

case 6:

checkout();

break;

case 7:

System.out.println("👋 Thank you for choosing " + hotelName + "!");

break;

default:

System.out.println("❌ Invalid choice.");

}

} while (choice != 7);

}

static void initializeRooms() {

rooms.add(new Room(101, "Standard", 1500));

rooms.add(new Room(102, "Standard", 1500));

rooms.add(new Room(201, "Deluxe", 2500));

rooms.add(new Room(202, "Deluxe", 2500));

rooms.add(new Room(301, "Suite", 4000));

rooms.add(new Room(302, "Suite", 4000));

}

static void viewAvailableRooms() {

System.out.println("\n--- Available Rooms ---");

boolean anyAvailable = false;

for (Room room : rooms) {

if (!room.isBooked) {

room.display();

anyAvailable = true;

}

}

if (!anyAvailable) {

System.out.println("All rooms are currently booked.");

}

}

static void bookRoom() {

System.out.print("Enter your name: ");

String name = sc.nextLine();

System.out.print("Enter your phone number: ");

String phone = sc.nextLine();

viewAvailableRooms();

System.out.print("Enter room number to book: ");

int roomNo = sc.nextInt();

sc.nextLine();

Room selected = null;

for (Room r : rooms) {

if (r.number == roomNo) {

selected = r;

break;

}

}

if (selected == null || selected.isBooked) {

System.out.println("❌ Room not available.");

return;

}

System.out.print("Enter number of nights: ");

int nights = sc.nextInt();

sc.nextLine();

double total = nights \* selected.pricePerNight;

String bookingId = UUID.randomUUID().toString().substring(0, 8);

selected.isBooked = true;

Booking booking = new Booking(bookingId, name, phone, roomNo, nights, total);

bookings.add(booking);

System.out.println("✅ Booking Confirmed!");

System.out.println("Booking ID: " + bookingId);

System.out.println("Total amount: ₹" + total);

}

static void cancelBooking() {

System.out.print("Enter your Booking ID to cancel: ");

String id = sc.nextLine();

Booking toCancel = null;

for (Booking b : bookings) {

if (b.bookingId.equals(id) && !b.isCheckedOut) {

toCancel = b;

break;

}

}

if (toCancel != null) {

bookings.remove(toCancel);

for (Room r : rooms) {

if (r.number == toCancel.roomNumber) {

r.isBooked = false;

break;

}

}

System.out.println("✅ Booking cancelled.");

} else {

System.out.println("❌ Active booking not found.");

}

}

static void checkout() {

System.out.print("Enter your Booking ID to checkout: ");

String id = sc.nextLine();

Booking current = null;

for (Booking b : bookings) {

if (b.bookingId.equals(id) && !b.isCheckedOut) {

current = b;

break;

}

}

if (current != null) {

current.isCheckedOut = true;

for (Room r : rooms) {

if (r.number == current.roomNumber) {

r.isBooked = false;

break;

}

}

System.out.println("✅ Checkout completed for Booking ID: " + id);

System.out.print("Would you like to leave a review? (yes/no): ");

String reply = sc.nextLine().trim().toLowerCase();

if (reply.equals("yes")) {

System.out.print("Write your review: ");

String comment = sc.nextLine();

reviews.add(new Review(current.guestName, comment));

System.out.println("✅ Thank you for your feedback!");

}

} else {

System.out.println("❌ Active booking not found.");

}

}

static void viewBookings() {

if (bookings.isEmpty()) {

System.out.println("No bookings found.");

} else {

System.out.println("\n--- Booking Details ---");

for (Booking b : bookings) {

b.display();

}

}

}

static void viewReviews() {

System.out.println("\n--- Guest Reviews ---");

if (reviews.isEmpty()) {

System.out.println("No reviews available yet.");

} else {

for (Review r : reviews) {

r.display();

}

}

}

}