GRAND HOTEL RESERVATION SYSTEM

This document contains the complete source code for the Grand Hotel Reservation System developed using Java, JDBC, and MySQL. The system allows users to search and book hotel rooms based on number of guests.

Technologies Used

Java, MySQL, JDBC

Room Features

Up to 2 Guests: WiFi, TV, Mini Bar, Room Service
Up to 3 Guests: WiFi, TV, Mini Bar, Work Desk
Up to 4 Guests: WiFi, TV, Mini Bar, Butler Service

MySQL Database Code

```
CREATE DATABASE grand_hotel;
USE grand_hotel;

CREATE TABLE rooms (
    room_id INT AUTO_INCREMENT PRIMARY KEY,
    room_type VARCHAR(50),
    max_guests INT,
    features VARCHAR(255),
    price DECIMAL(10,2)
);

CREATE TABLE bookings (
    booking_id INT AUTO_INCREMENT PRIMARY KEY,
    check_in DATE,
    check_out DATE,
    guests INT,
    room_type VARCHAR(50)
);

INSERT INTO rooms VALUES
(1,'Deluxe Room',2,'WiFi, TV, Mini Bar, Room Service',2500),
(2,'Executive Room',3,'WiFi, TV, Mini Bar, Work Desk',3500),
(3,'Luxury Suite',4,'WiFi, TV, Mini Bar, Butler Service',5000);
```

DBConnection.java

```
import java.sql.Connection;
import java.sql.DriverManager;

public class DBConnection {
    public static Connection getConnection() {
        Connection con = null;
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/grand_hotel",
                "root",
                "password"
            );
        } catch (Exception e) {
            e.printStackTrace();
        }
        return con;
    }
}
```

RoomDAO.java

```
import java.sql.*;

public class RoomDAO {
    public static void searchRooms(int guests) {
        try {
            Connection con = DBConnection.getConnection();
            PreparedStatement ps =
                con.prepareStatement("SELECT * FROM rooms WHERE max_guests >= ?");
            ps.setInt(1, guests);
            ResultSet rs = ps.executeQuery();

            while(rs.next()) {
                System.out.println("Room Type: " + rs.getString("room_type"));
                System.out.println("Features: " + rs.getString("features"));
                System.out.println("Price: Rs." + rs.getDouble("price"));
                System.out.println("------------------------");
            }
        } catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

BookingDAO.java

```
import java.sql.*;

public class BookingDAO {
    public static void bookRoom(String checkIn, String checkOut, int guests, String roomType) {
        try {
            Connection con = DBConnection.getConnection();
            PreparedStatement ps =
                con.prepareStatement(
                "INSERT INTO bookings(check_in, check_out, guests, room_type) VALUES (?,?,?,?)");

            ps.setString(1, checkIn);
            ps.setString(2, checkOut);
            ps.setInt(3, guests);
            ps.setString(4, roomType);

            ps.executeUpdate();
            System.out.println("Room booked successfully!");
        } catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

HotelReservation.java (Main Program)

```
import java.util.Scanner;

public class HotelReservation {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Welcome to GRAND HOTEL");
        System.out.print("Enter Check-in Date (YYYY-MM-DD): ");
        String checkIn = sc.next();

        System.out.print("Enter Check-out Date (YYYY-MM-DD): ");
        String checkOut = sc.next();

        System.out.print("Enter Number of Guests (2/3/4): ");
        int guests = sc.nextInt();

        RoomDAO.searchRooms(guests);

        sc.nextLine();
        System.out.print("Enter Room Type to Book: ");
        String roomType = sc.nextLine();

        BookingDAO.bookRoom(checkIn, checkOut, guests, roomType);
        sc.close();
    }
}
```