

# OOPS MINI PROJECT

## ROOM BOOKING MANAGEMENT SYSTEM

### AIM:

To develop a Room Management System that connect to a My SQL database and allow users to:

### Algorithm:

1 Start

2 Connect to the MySQL database.

3 Display a menu with the following options:

- Add Room
- View Available Rooms
- Book Room
- View Bookings
- Exit

4 Based on the user's choice:

- If Add Room:
  - Take input: Room Type, Room Number, Price, and Availability.
  - Insert data into the Rooms table.
- If View Available Rooms:
  - Display rooms that are available.
- If Book Room:
  - Take input: Customer Name, Room Number, and Booking Date.

- Insert booking details into the Bookings table.
- If View Bookings:
  - Display all current bookings.
- If Exit:
  - Close the database connection and end the program.

5 Repeat the process until the user chooses to exit.

6 Stop.

## **SQL Queries:**

### **1.Create Database:**

```
CREATE DATABASE RoomManagement;
```

### **2.Use the database:**

```
USE RoomManagement;
```

### **3.Create Rooms Table**

```
CREATE TABLE Rooms (  
    room_id INT PRIMARY KEY AUTO_INCREMENT,  
    room_type VARCHAR(50),  
    room_number VARCHAR(20),  
    price DECIMAL(10, 2),  
    is_available BOOLEAN DEFAULT TRUE  
);
```

### **4.Create Bookings Table:**

```
CREATE TABLE Bookings (  
    booking_id INT PRIMARY KEY AUTO_INCREMENT,  
    customer_name VARCHAR(100),
```

```
room_number VARCHAR(20),  
booking_date DATE,  
FOREIGN KEY (room_number) REFERENCES Rooms(room_number)  
);
```

### **Java Code:**

```
import java.sql.*;  
import java.util.Scanner;  
  
public class RoomManagementSystem {  
  
    private static final String URL =  
"jdbc:mysql://localhost:3306/RoomManagement";  
    private static final String USER = "root";  
    private static final String PASSWORD = "";  
  
    public static void main(String[] args) {  
        try (Connection conn = DriverManager.getConnection(URL, USER,  
PASSWORD);  
            Scanner scanner = new Scanner(System.in)) {  
            System.out.println("Connected to the database!");  
  
            while (true) {  
                System.out.println("\n--- Room Management System ---");  
                System.out.println("1. Add Room");
```

```

        System.out.println("2. View Available Rooms");
        System.out.println("3. Book Room");
        System.out.println("4. View Bookings");
        System.out.println("5. Exit");
        System.out.print("Choose an option: ");
        int choice = scanner.nextInt();
        scanner.nextLine(); // Consume newline

        switch (choice) {
            case 1 -> addRoom(conn, scanner);
            case 2 -> viewAvailableRooms(conn);
            case 3 -> bookRoom(conn, scanner);
            case 4 -> viewBookings(conn);
            case 5 -> {
                System.out.println("Exiting...");
                return;
            }
            default -> System.out.println("Invalid choice. Try again.");
        }
    }
} catch (SQLException e) {
    e.printStackTrace();
}
}

```

```

private static void addRoom(Connection conn, Scanner scanner) throws
SQLException {

```

```
System.out.print("Enter Room Type: ");  
String roomType = scanner.nextLine();  
System.out.print("Enter Room Number: ");  
String roomNumber = scanner.nextLine();  
System.out.print("Enter Price: ");  
double price = scanner.nextDouble();  
scanner.nextLine(); // Consume newline
```

```
String query = "INSERT INTO Rooms (room_type, room_number, price,  
is_available) VALUES (?, ?, ?, TRUE)";
```

```
try (PreparedStatement stmt = conn.prepareStatement(query)) {  
    stmt.setString(1, roomType);  
    stmt.setString(2, roomNumber);  
    stmt.setDouble(3, price);  
    stmt.executeUpdate();  
    System.out.println("Room added successfully!");  
}  
}
```

```
private static void viewAvailableRooms(Connection conn) throws  
SQLException {
```

```
String query = "SELECT * FROM Rooms WHERE is_available = TRUE";  
try (Statement stmt = conn.createStatement();  
    ResultSet rs = stmt.executeQuery(query)) {  
    System.out.println("\n--- Available Rooms ---");  
    while (rs.next()) {
```

```

        System.out.println("Room Number: " + rs.getString("room_number")
+
        ", Room Type: " + rs.getString("room_type") +
        ", Price: $" + rs.getDouble("price"));
    }
}
}

```

```

private static void bookRoom(Connection conn, Scanner scanner) throws
SQLException {
    System.out.print("Enter Customer Name: ");
    String customerName = scanner.nextLine();
    System.out.print("Enter Room Number: ");
    String roomNumber = scanner.nextLine();

    // Check if room is available
    String checkQuery = "SELECT is_available FROM Rooms WHERE
room_number = ?";
    try (PreparedStatement checkStmt = conn.prepareStatement(checkQuery))
    {
        checkStmt.setString(1, roomNumber);
        ResultSet rs = checkStmt.executeQuery();
        if (rs.next() && rs.getBoolean("is_available")) {
            // Proceed with booking
            String bookingQuery = "INSERT INTO Bookings (customer_name,
room_number, booking_date) VALUES (?, ?, NOW())";
            try (PreparedStatement stmt =
conn.prepareStatement(bookingQuery)) {

```

```

        stmt.setString(1, customerName);
        stmt.setString(2, roomNumber);
        stmt.executeUpdate();

        // Update room availability

        String updateRoomQuery = "UPDATE Rooms SET is_available = FALSE
WHERE room_number = ?";

        try (PreparedStatement updateStmt =
conn.prepareStatement(updateRoomQuery)) {
            updateStmt.setString(1, roomNumber);
            updateStmt.executeUpdate();
        }

        System.out.println("Room booked successfully!");
    }
} else {
    System.out.println("Room is not available.");
}
}
}

private static void viewBookings(Connection conn) throws SQLException {
    String query = "SELECT * FROM Bookings";
    try (Statement stmt = conn.createStatement();
        ResultSet rs = stmt.executeQuery(query)) {
        System.out.println("\n--- All Bookings ---");
        while (rs.next()) {

```

```
        System.out.println("Booking ID: " + rs.getInt("booking_id") +  
            ", Customer Name: " + rs.getString("customer_name") +  
            ", Room Number: " + rs.getString("room_number") +  
            ", Booking Date: " + rs.getDate("booking_date"));  
    }  
}  
}
```

## Input:

--- Room Management System ---

1. Add Room
2. View Available Rooms
3. Book Room
4. View Bookings
5. Exit

Choose an option:

### **1.Add Room:**

Enter Room Type:

Enter Room Number:

Enter Price:



## 2. View Available Rooms:

--- Available Rooms ---

Room Number: 101, Room Type: Deluxe, Price: \$150.00

## 3. Book Room:

Enter Customer Name:

Enter Room Number:

## 4. View Bookings:

--- All Bookings ---

Booking ID: 1,

Customer Name: medvin deva,

Room Number: 101,

Booking Date: 2024-03-01

## 5. Exit:

Exiting...

# OUTPUT

## 1.Add Room

```
Enter Room Type: Deluxe
Enter Room Number: 101
Enter Price: 150.00
Room added successfully!
```

## 2.View Available Bikes

```
Room Number: 101, Room Type: Deluxe, Price: $150.00
```

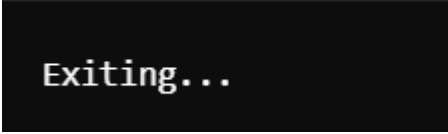
## 3.Booking a Room

```
Enter Customer Name: John Doe
Enter Room Number: 101
Room booked successfully!
```

## 4. Viewing Bookings:

```
Booking ID: 1, Customer Name: John Doe, Room Number: 101, Booking Date: 2024-03-01
```

## 5.Exiting:



```
Exiting...
```

## RESULT:

THE DATABASE CONSTRUCTION FOR THE ROOM BOOKING MANAGEMENT SYSTEM HAS BEEN SUCCESSFULLY COMPLETED AND CONNECTED WITH SQL USING JAVA.