

## **About the Project**

**(This was done and documented in 2019 by  
Madhav Sirohi)**

The project is all about a Hotel Management System which can be used to do bookings, maintain them, editing them and simultaneously handling customer/client details. This system can be used for offline and low-level data entry (preferably for a hotel with 100 clients per month at most). This project consists of Java front-end (developed by NetBeans IDE 8.2) and MySQL back end.

# Front-End

## **Contents:**

- 1.Prerequisites for Database Connectivity
- 2.Log-In Frame
- 3.Menu Frame
- 4.View Bookings Frame
- 5.Edit Bookings Frame
- 6.Add Bookings Frame
- 7.View Customer Details Frame
- 8.Add Customer Details Frame
- 9.Generate Bill Frame

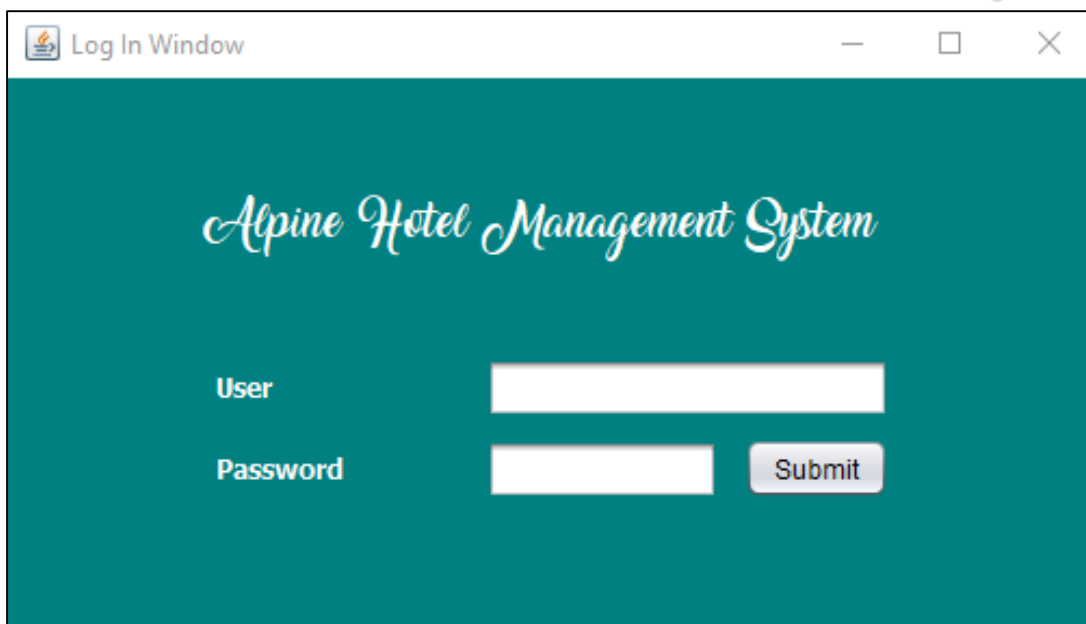
## **Prerequisites for Database Connectivity**

Import JAR files for MySQL Connector into the project and on the top of View Bookings Frame, Edit Bookings Frame, View Customer Details Frame, and Edit Customer Details Frame Source Code Editor add the following code:

```
import java.sql.*;  
import javax.swing.table.DefaultTableModel;  
import javax.swing.JOptionPane;
```

# Log-In Frame

## Screenshot:



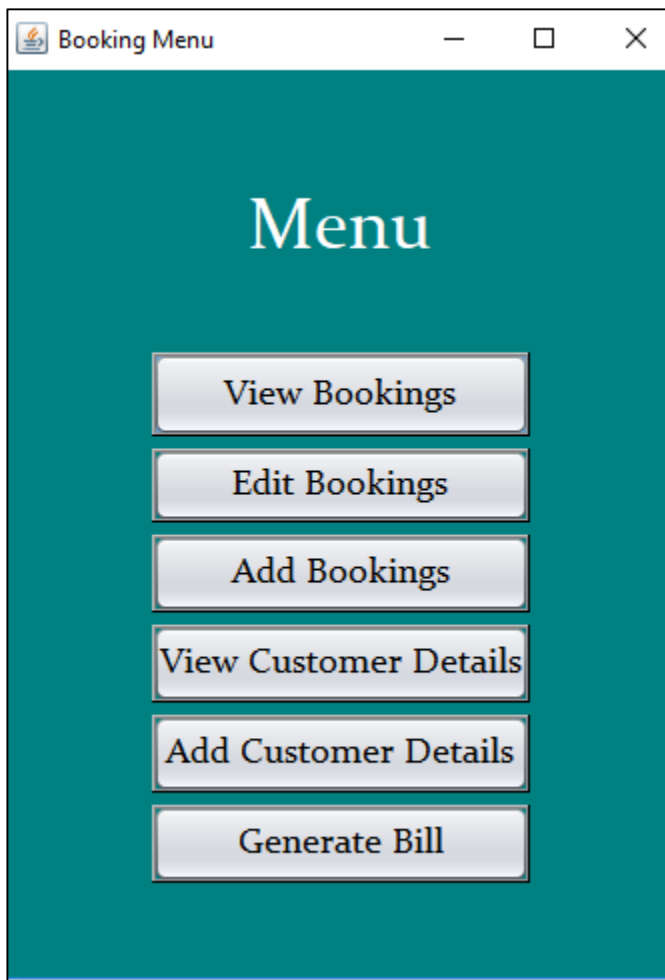
## Source Code:

- Add the following code in Submit Button's Action Performed Method:

```
String user=new String(jPasswordField1.getPassword());
String pass=new String(jPasswordField2.getPassword());
if(user.equals("Admin") && pass.equals("Password"))
{
    new LogIn().setVisible(false);
    JOptionPane.showMessageDialog(null,"Welcome Admin!");
    new Menu().setVisible(true);
}
else
{
    JOptionPane.showMessageDialog(null,"Incorrect Details.");
}
```

## Menu Frame

### Screenshot:



### Source Code:

- Add the following code in View Bookings Button's Action Performed Method:

```
new View_Bookings().setVisible(true);
```

- Add the following code in Edit Bookings Button's Action Performed Method:

```
new Edit_Bookings().setVisible(true);
```

- Add the following code in Add Bookings Button's Action Performed Method:

```
new AddBookings().setVisible(true);
```

- Add the following code in View Customer Details Button's Action Performed Method:

```
new View_Customer_Details().setVisible(true);
```

- Add the following code in Add Customer Details Button's Action Performed Method:

```
new Add_Customer_Details().setVisible(true);
```

- Add the following code in Generate Bill Button's Action Performed Method:

```
new Bill_Generator().setVisible(true);
```

# View Bookings Frame

## Screenshot:

Specify Search Criteria below

Search Field: Booking\_date = Criteria(YYYY-MM-DD):

Customer Name	Booking Date	Check In Date	Check Out Date	Address
---------------	--------------	---------------	----------------	---------

Buttons: Clear Table, View all Records(Irrespective of all Criteria), Search in the Database, Return to Menu

## Source Code:

- Add the following code in Clear Table Button's Action Performed Method:

```
DefaultTableModel model=(DefaultTableModel)Bookings.getModel();
int rows=model.getRowCount();
if(rows>0)
{
    for(int i=0;i<rows;i++)
```

```

        {
            model.removeRow(0);
        }
    }
}

```

- Add the following code in View all Records Button's Action Performed Method:

```

emptyBtn.doClick();
DefaultTableModel
model=(DefaultTableModel)Bookings.getModel();
try {
    Class.forName("java.sql.Driver");
    Connection con=
DriverManager.getConnection("jdbc:mysql://localhost/Alpine_Hotel","root",
"*****");
    Statement stmt=con.createStatement();
    String query="select
Customer_name,Booking_date,CheckIn_date,CheckOut_date,Address from
Bookings;"; //Required Query
    ResultSet rs=stmt.executeQuery(query);
    while(rs.next())
    {
        String nm=rs.getString(1);
        String bd=rs.getString(2);
        String cid= rs.getString(3);
        String cod= rs.getString(4);
        String add= rs.getString(5);
        model.addRow(new Object[]{nm,bd,cid,cod,add});
    }
    rs.close();
    stmt.close();
    con.close();
}
catch(Exception e)
{
    JOptionPane.showMessageDialog(null,"Error in
Connectivity.");
}
}

```

- Add the following code in Search in Database Button's Action Performed Method:

```

emptyBtn.doClick();
DefaultTableModel
model=(DefaultTableModel)Bookings.getModel();
try {
    Class.forName("java.sql.Driver");
    Connection con=
DriverManager.getConnection("jdbc:mysql://localhost/Alpine_Hotel","root",
"*****");
    Statement stmt=con.createStatement();
    String sfld=(String)srchFldCBX.getSelectedItem();
    String op=(String) opCbx.getSelectedItem();
    String crit=criteriaTF.getText();
}

```



```

        String query="select
Customer_name,Booking_date,CheckIn_date,CheckOut_date,Address from
Bookings "+"where "+sfld+op+crit+""; //Required Query
        ResultSet rs=stmt.executeQuery(query);
        while(rs.next())
        {
            String nm=rs.getString(1);
            String bd=rs.getString(2);
            String cid= rs.getString(3);
            String cod= rs.getString(4);
            String add= rs.getString(5);
            model.addRow(new Object[]{nm,bd,cid,cod,add});
        }
        rs.close();
        stmt.close();
        con.close();
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(null,"Error in
Connectivity.");
    }
}

```

- Add the following code in Return to Menu Button's Action Performed Method:

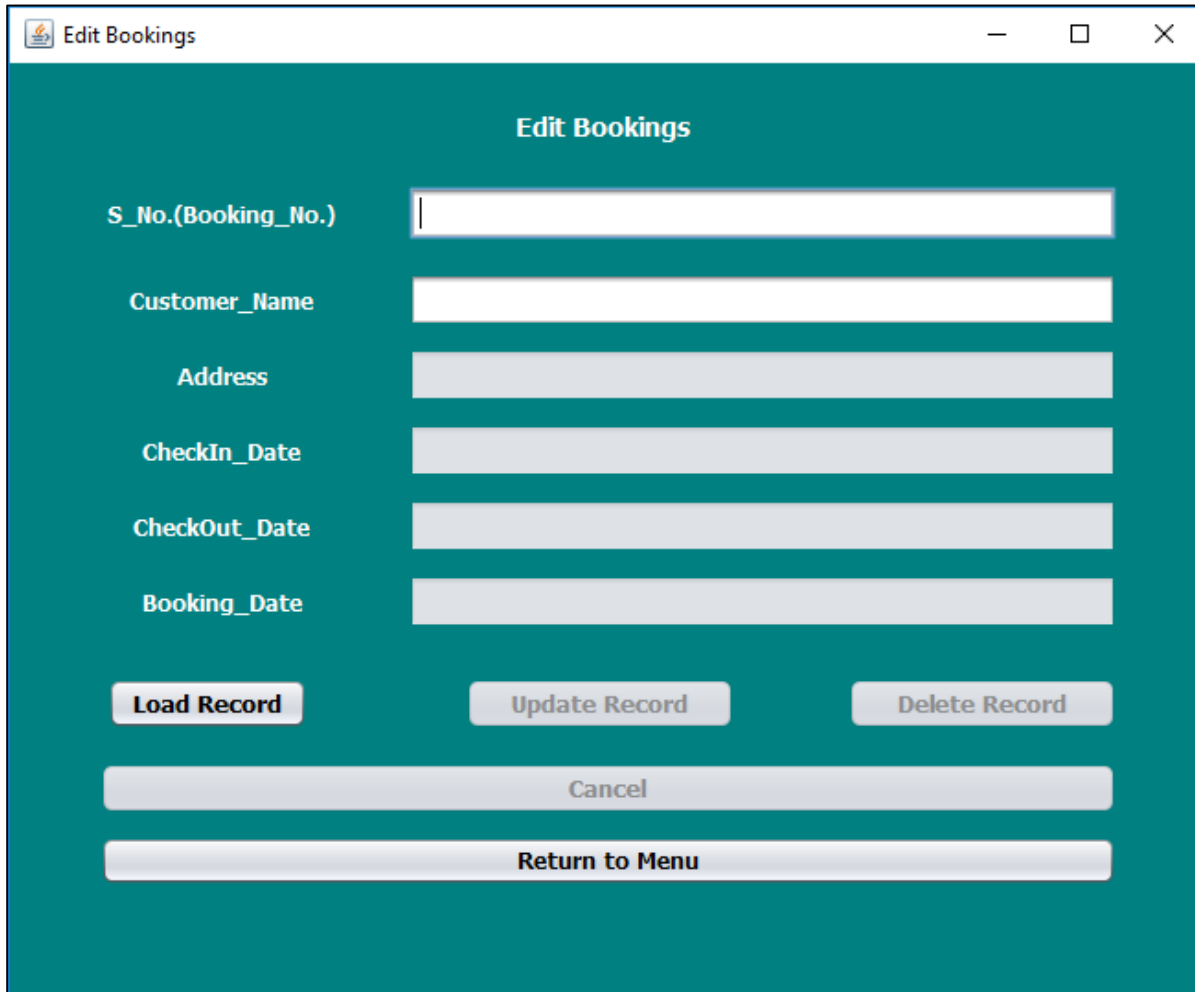
```

new Menu().setVisible(true);

```

# Edit Bookings Frame

Screenshot:

A screenshot of a Java Swing window titled "Edit Bookings". The window has a teal background and a white title bar with standard window controls. The main content area contains a form with the following fields: "S\_No.(Booking\_No.)", "Customer\_Name", "Address", "CheckIn\_Date", "CheckOut\_Date", and "Booking\_Date". Each field is followed by a text input box. Below the input boxes are five buttons: "Load Record", "Update Record", "Delete Record", "Cancel", and "Return to Menu". The buttons are arranged in two rows: the first row contains "Load Record", "Update Record", and "Delete Record"; the second row contains "Cancel" and "Return to Menu".

Source Code:

- Add the following code in the 'public class Edit\_Bookings extends javax.swing.JFrame{' Method(on the top):

```
Connection con=null;
Statement stmt=null;
ResultSet rs=null;
```

- Add the following code in Load Record Button's Action Performed Method:

```

try {
    Class.forName("java.sql.Driver");
    con=
DriverManager.getConnection("jdbc:mysql://localhost/Alpine_Hotel","root", "*****");
    stmt=con.createStatement();
    String query="select * from Bookings where
Customer_Name=\""+cntf.getText()+"\" && S_No="+sntf.getText()+" ";
//Required Query
rs=stmt.executeQuery(query);
if(rs.next())
{
    String Add=rs.getString(3);
    String bd= rs.getString(4);
    String cid= rs.getString(5);
    String cod= rs.getString(6);
    upBtn.setEnabled(true);
    DlBtn.setEnabled(true);
    cidtf.setEnabled(true);
    codtf.setEnabled(true);
    sntf.setEnabled(false);
    cntf.setEnabled(false);
    jButton1.setEnabled(true);
    cidtf.setText(cid);
    codtf.setText(cod);
    bdtf.setText(bd);
    Addtf.setText(Add);
}
else
{
    JOptionPane.showMessageDialog(null,"No such Record
found.");
}
}
catch(Exception e)
{
    JOptionPane.showMessageDialog(null,"Error in
Connectivity.");
}
}

```

- Add the following code in Update Record Button's Action Performed Method:

```

try {
    stmt=con.createStatement();
    String query="update Bookings set
CheckIn_date='"+cidtf.getText()+"',
CheckOut_date='"+codtf.getText()+"'"+where
Customer_Name=\""+cntf.getText()+"\" && S_No="+sntf.getText()+" ";
//Required Query

```

```

        stmt.executeUpdate(query);
        JOptionPane.showMessageDialog(null,"Record successfully
updated.");
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(null,"Error in Table
updatation.");
    }
    sntf.setText("");
    cntf.setText("");
    Addtf.setText("");
    cidtf.setText("");
    codtf.setText("");
    bdtf.setText("");
    upBtn.setEnabled(false);
    DlBtn.setEnabled(false);

```

- Add the following code in Delete Record Button's Action Performed Method:

```

    int res=JOptionPane.showConfirmDialog(null,"Do you want to delete
the Record?");
    if(res==JOptionPane.YES_OPTION)
    {
        try
        {
            stmt=con.createStatement();
            String query="delete from Bookings where
Customer_Name=\""+cntf.getText()+"\" && S_No="+sntf.getText()+" ";
//Required Query
            stmt.executeUpdate(query);
            JOptionPane.showMessageDialog(null,"Record deleted.");
            sntf.setText("");
            cntf.setText("");
            Addtf.setText("");
            cidtf.setText("");
            codtf.setText("");
            bdtf.setText("");
            upBtn.setEnabled(false);
            DlBtn.setEnabled(false);
            sntf.setEnabled(true);
            cntf.setEnabled(true);
            Addtf.setEnabled(false);
            cidtf.setEnabled(false);
            codtf.setEnabled(false);
            bdtf.setEnabled(false);
            jButton1.setEnabled(false);
        }
        catch(Exception e)
        {
            JOptionPane.showMessageDialog(null,"Error in
deletion.");
        }
    }
}

```

- Add the following code in Return to Menu Button's Action Performed Method:

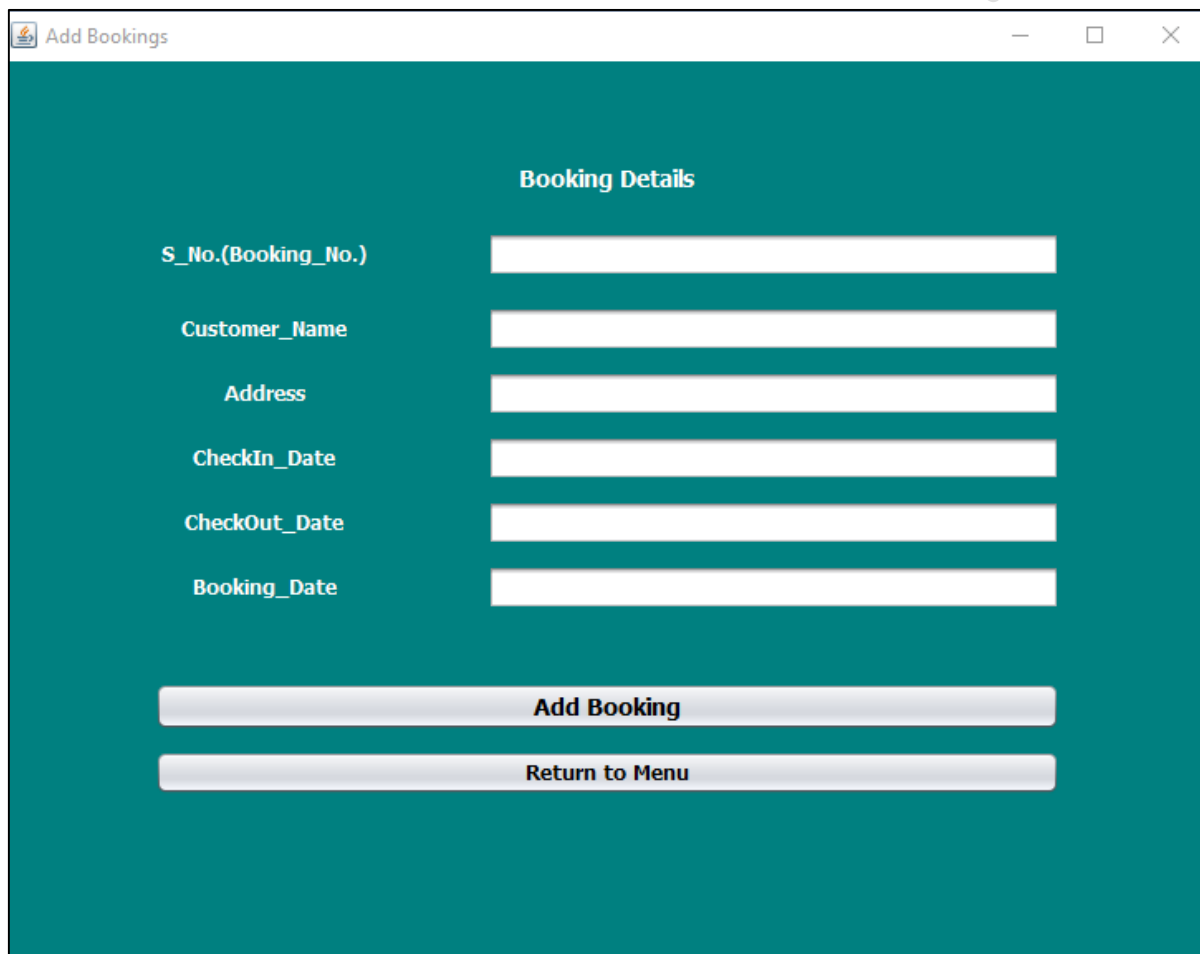
```
new Menu().setVisible(true);
```

- Add the following code in Cancel Button's Action Performed Method:

```
sntf.setText("");  
cntf.setText("");  
Addtf.setText("");  
cidtf.setText("");  
codtf.setText("");  
bdtf.setText("");  
sntf.setEnabled(true);  
cntf.setEnabled(true);  
Addtf.setEnabled(false);  
cidtf.setEnabled(false);  
codtf.setEnabled(false);  
bdtf.setEnabled(false);  
DlBtn.setEnabled(false);  
upBtn.setEnabled(false);  
jButton1.setEnabled(false);
```

## Add Bookings Frame

### Screenshot:



The screenshot shows a window titled "Add Bookings" with a teal background. The window contains a form titled "Booking Details" with the following fields:

Booking Details	
S_No.(Booking_No.)	<input type="text"/>
Customer_Name	<input type="text"/>
Address	<input type="text"/>
CheckIn_Date	<input type="text"/>
CheckOut_Date	<input type="text"/>
Booking_Date	<input type="text"/>

Below the form are two buttons:

- Add Booking**
- Return to Menu**

### Source Code:

- Add the following code in Add Bookings Button's Action Performed Method:

```
try {  
    Class.forName("java.sql.Driver");  
    Connection con=  
DriverManager.getConnection("jdbc:mysql://localhost/Alpine_Hotel","root", "*****");  
    Statement stmt=con.createStatement();
```

```

        String query="insert into Bookings values ("
            +sntf.getText()+",' '
            +cntf.getText()+",' '
            +Addtf.getText()+",' '
            +bdtf.getText()+",' '
            +cidtf.getText()+",' '
            +codtf.getText()+''"+");"; //Required Query
        stmt.executeUpdate(query);
        JOptionPane.showMessageDialog(null,"Booking
successfully done.");
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(null,"Error in Booking.");
    }
}

```

- Add the following code in Return to Menu Button's Action Performed Method:

```

new Menu().setVisible(true);

```

## View Customer Details Frame

### Screenshot:

Customer Details

Specify Search Criteria below

Search Field Criteria

S\_No =

S_No	No of People	Room No	Type of Room	Payment Due
------	--------------	---------	--------------	-------------

Clear Table View all Records(Irrespective of all Criteria) Search in the Database

Return to Menu

### Source Code:

- Add the following code in Clear Table Button's Action Performed Method:

```
DefaultTableModel model=(DefaultTableModel)Cus.getModel();
int rows=model.getRowCount();
if(rows>0)
{
    for(int i=0;i<rows;i++)
    {
        model.removeRow(0);
    }
}
```



```

    }
}

```

- Add the following code in View all Records Button's Action Performed Method:

```

emptyBtn.doClick();
DefaultTableModel model=(DefaultTableModel)Cus.getModel();
try {
    Class.forName("java.sql.Driver");
    Connection con=
DriverManager.getConnection("jdbc:mysql://localhost/Alpine_Hotel","root",
"*****");
    Statement stmt=con.createStatement();
    String query="select * from customer_details;";
//Required Query
    ResultSet rs=stmt.executeQuery(query);
    while(rs.next())
    {
        String sn=rs.getString(1);
        String nop=rs.getString(2);
        String rn= rs.getString(3);
        String tor= rs.getString(4);
        String pd= rs.getString(5);
        model.addRow(new Object[]{sn,nop,rn,tor,pd});
    }
    rs.close();
    stmt.close();
    con.close();
}
catch(Exception e)
{
    JOptionPane.showMessageDialog(null,"Error in
Connectivity.");
}

```

- Add the following code in Search in Database Button's Action Performed Method:

```

emptyBtn.doClick();
DefaultTableModel model=(DefaultTableModel)Cus.getModel();
try {
    Class.forName("java.sql.Driver");
    Connection con=
DriverManager.getConnection("jdbc:mysql://localhost/Alpine_Hotel","root",
"*****");
    Statement stmt=con.createStatement();
    String sfld=(String)srchFldCBX.getSelectedItem();
    String op=(String) opCbx.getSelectedItem();
    String crit=criteriaTF.getText();
    String query="select * from customer_details "+"where
"+sfld+op+crit+""; //Required Query
    ResultSet rs=stmt.executeQuery(query);
    while(rs.next())
    {

```

```

        String sn=rs.getString(1);
        String nop=rs.getString(2);
        String rn= rs.getString(3);
        String tor= rs.getString(4);
        String pd= rs.getString(5);
        model.addRow(new Object[]{sn,nop,rn,tor,pd});
    }
    rs.close();
    stmt.close();
    con.close();
}
catch(Exception e)
{
    JOptionPane.showMessageDialog(null,"Error in
Connectivity.");
}

```

- Add the following code in Return to Menu Button's Action Performed Method:

```
new Menu().setVisible(true);
```

## Add Customer Details Frame

Screenshot:

**Customer Details**

S\_No.(Booking\_No.)

Number of People

Room NO

Type of Room

Payment Due

**Add Details**

**Return to Menu**

Source Code:

- Add the following code in Add Bookings Button's Action Performed Method:

```
try {  
    Class.forName("java.sql.Driver");  
    Connection con=  
DriverManager.getConnection("jdbc:mysql://localhost/Alpine_Hotel","root","*****");  
    Statement stmt=con.createStatement();  
    String query="insert into customer_details values ("
```

```

+sntf.getText()+"", "
+nptf.getText()+"", "
+rintf.getText()+"', "
+(String)jComboBox1.getSelectedItem()+"', "
+pdtf.getText()+"");"; //Required Query
stmt.executeUpdate(query);
JOptionPane.showMessageDialog(null,"Details successfully
added.");
}
catch(Exception e)
{
    JOptionPane.showMessageDialog(null,"Error in adding.");
}

```

- Add the following code in Return to Menu Button's Action Performed Method:

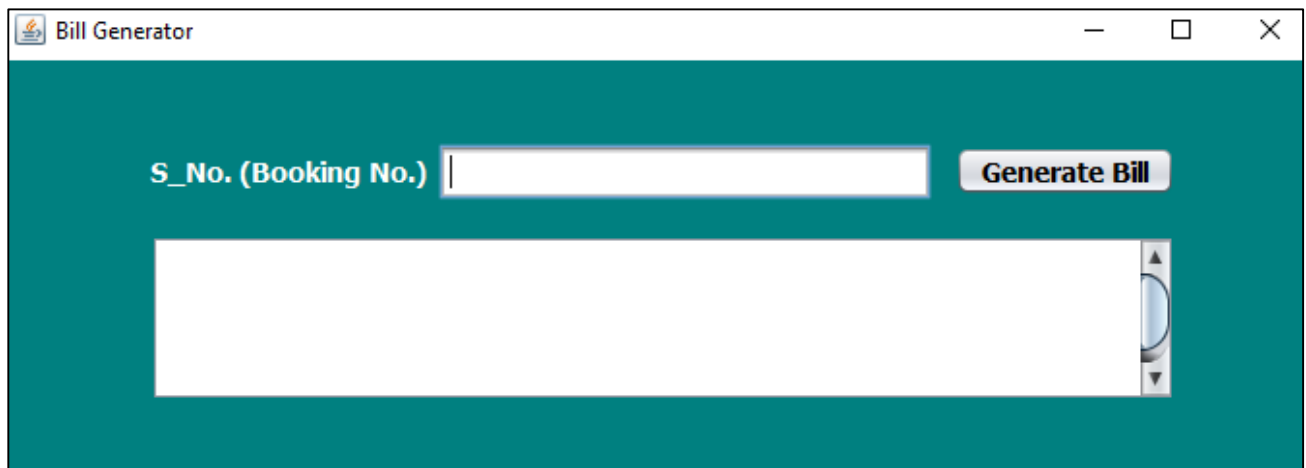
```

new Menu().setVisible(true);

```

# Generate Bill Frame

## Screenshot:



## Source Code:

- Add the following code in Generate Bill Button's Action Performed Method:

```
try {
    Class.forName("java.sql.Driver");
    Connection con=
    DriverManager.getConnection("jdbc:mysql://localhost/Alpine_Hotel","root","*****");
    Statement stmt=con.createStatement();
    String query="select * from customer_details "+"where
    S_No="+jTextField1.getText()+" "; //Required Query
    ResultSet rs=stmt.executeQuery(query);
    while(rs.next())
    {
        String nop=rs.getString(2);
        String tor= rs.getString(4);
        String pd= rs.getString(5);
        switch(tor)
        {
            case "NON-AC":jTextArea1.setText("The Payable
            amountis Rs"+(Integer.parseInt(nop)*100+500)+". "+" For "+nop+" People
            and NON-AC Room.");break;
            case "AC":jTextArea1.setText("The Payable amountis
            Rs"+(Integer.parseInt(nop)*100+700)+". "+" For "+nop+" People and AC
            Room.");break;
```

```
        case "DELUXE":jTextArea1.setText("The Payable  
amountis Rs"+(Integer.parseInt(nop)*100+1500)+"."+ " For "+nop+" People  
and Deluxe Room.");break;  
        default: jTextArea1.setText("The Payable amountis  
Rs"+(Integer.parseInt(nop)*100+500)+"."+ " For "+nop+" People and NON-  
AC Room.");  
    }  
}  
rs.close();  
stmt.close();  
con.close();  
}  
catch(Exception e)  
{  
    JOptionPane.showMessageDialog(null,"Error in  
Connectivity.");  
}
```

# Back-End

## Contents:

A MySQL database named Alpine\_Hotel containing 2 tables viz. Bookings and Customer\_Details.

## Table Bookings structure:

```
mysql> desc Bookings;
```

Field	Type	Null	Key	Default	Extra
S_No	int(11)	NO	PRI	NULL	
Customer_Name	varchar(50)	NO		NULL	
Address	varchar(200)	NO		NULL	
Booking_date	date	NO		NULL	
CheckIn_date	date	NO		NULL	
CheckOut_date	date	NO		NULL	

6 rows in set (1.14 sec)

## Table Customer\_Details structure:

```
mysql> desc Customer_Details;
```

Field	Type	Null	Key	Default	Extra
S_No	int(11)	YES	MUL	NULL	
No_of_People	int(11)	NO		NULL	
Room_No	int(11)	NO		NULL	
Type_of_Room	varchar(30)	NO		NULL	
Payment_Due	float	NO		NULL	

5 rows in set (0.00 sec)

## **Merits:**

- Fully Automated.
- Neat and clean Data-Entry without Repudiation and Repetition as MySQL is an RDBMS.
- Cost-effective, saves Paper and Manual Labour.
- Increases efficiency.
- One click and a quick search through the Database.



## **Demerits:**

- Not very intuitive (easy to understand) interface, at least some basic knowledge is required.
- Chances of Complete Data Loss because data-entry is offline.

## **Bibliography:**

- Class XII IP Textbook by Sumita Arora.
- Search Engine: Google.
- Wikipedia

Thank  
You