Availability.Java

import java.util.\*;

import java.sql.\*;

import java.text.SimpleDateFormat;

import javax.swing.JTable;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.AbstractTableModel;

public class Availability extends javax.swing.JFrame {

/\*\* Creates new form Availability \*/

int ageDays;

Connection con;

Statement stmt,smt1;

ResultSet rs;

java.util.Date dt1,dt2;

public Object obj1,obj2,obj3,obj4,obj5,obj6;

String diffday;

public Availability() {

initComponents();

try

{

Class .forName("com.mysql.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/hotel","root","");

jDateChooser1.setDate(Calendar.getInstance().getTime());

}

catch(Exception e)

{

System.out.println(e.getMessage());

}

}

static final long ONE\_HOUR = 60 \* 60 \* 1000L;

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton3ActionPerformed

this.dispose();

}//GEN-LAST:event\_jButton3ActionPerformed

//CHECKS THE RROM AVAILIBILITY AFTER ENTERING DATES

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton2ActionPerformed

final Object[] columnNames=new String[] {"Room No.","Room Type","Bed Type","Tariff Per Room"};

SimpleDateFormat sdf= new SimpleDateFormat("yyyy-MM-dd");

DefaultTableModel dtm=new DefaultTableModel(columnNames,0);

dt1=jDateChooser1.getDate();

dt2=jDateChooser2.getDate();

String strdtver1=(String) sdf.format(jDateChooser1.getDate());

String strdtver2=(String) sdf.format(jDateChooser2.getDate());

long diff=((dt2.getTime()-dt1.getTime()+ONE\_HOUR)/(ONE\_HOUR\*24));

diffday=Long.toString(diff);

jTextField4.setText(diffday+" day(s)");

String sql="SELECT room\_no, room\_type, room\_bed, room\_rate FROM roomdetail "

+ "where room\_no NOT IN (SELECT room\_no FROM bookingtable WHERE DATE(date\_to) >= '"

+ strdtver1 + "' AND DATE(date\_fro) <= '" + strdtver2 + "');";

/\*"SELECT roomdetail.room\_no, room\_type, room\_bed, room\_rate "

+ "FROM roomdetail LEFT JOIN bookingtable "

+ "ON roomdetail.room\_no=bookingtable.room\_no AND DATE(date\_fro) >= '"

+ strdtver1 + "' AND DATE(date\_to) <= '" + strdtver2 + "' WHERE bookingtable.room\_no IS NULL;";\*/

try

{

smt1=con.createStatement(ResultSet.TYPE\_SCROLL\_SENSITIVE,ResultSet.CONCUR\_UPDATABLE);

rs = smt1.executeQuery(sql);

int i = 0;

while(rs.next())

{

String var1=rs.getString(1);

String var2=rs.getString(2);

String var3=rs.getString(3);

String var4=Integer.toString(rs.getInt(4));

dtm.addRow(new Vector());

dtm.setValueAt(var1, i, 0);

dtm.setValueAt(var2, i, 1);

dtm.setValueAt(var3, i, 2);

dtm.setValueAt(var4, i, 3);

i++;

}

jTable1.setModel(dtm);

/\*Object obj1 = GetData(jTable1, 2, 2);

System.out.println("Cell value of 3 column and 3 row :" + obj1);

Object obj2 = GetData(jTable1, 2, 1);

System.out.println("Cell value of 2 column and 3 row :" + obj2);\*/

}

catch(Exception ex)

{

System.out.println(ex.getMessage());

ex.printStackTrace();

}

}//GEN-LAST:event\_jButton2ActionPerformed

public Object GetData(JTable jTable1, int row\_index, int col\_index){

return jTable1.getModel().getValueAt(row\_index, col\_index);

}

//OPENS THE BOOKING FRAME FOR BOOKING CONFIRMATION

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton4ActionPerformed

BookRoom br=new BookRoom(obj1,obj2,obj3,obj4,dt1,dt2,diffday);

br.setVisible(true);

}//GEN-LAST:event\_jButton4ActionPerformed

private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {//GEN-FIRST:event\_jTable1MouseClicked

int rno=jTable1.getSelectedRow();

obj1 = GetData(jTable1, rno, 0);

obj2 = GetData(jTable1, rno, 1);

obj3 = GetData(jTable1, rno, 2);

obj4 = GetData(jTable1, rno, 3);

obj5=dt1;

obj6=dt2;

}//GEN-LAST:event\_jTable1MouseClicked

private void formWindowOpened(java.awt.event.WindowEvent evt) {//GEN-FIRST:event\_formWindowOpened

jPanel1.setOpaque(true);

jLabel9.setOpaque(true);

jTable1.setToolTipText("Select the Room want to reserve");

jButton4.setToolTipText("Select the Room from the list and click this button for Room Booking");

}//GEN-LAST:event\_formWindowOpened

private void formWindowClosing(java.awt.event.WindowEvent evt) {//GEN-FIRST:event\_formWindowClosing

}//GEN-LAST:event\_formWindowClosing

/\*\*

\* @param args the command line arguments

public static void main(String args[]) {

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new Availability().setVisible(true);

}

});

}

BookingMang.Java

import java.sql.\*;

public class BookingMang extends javax.swing.JFrame {

Connection con;

Statement smt,smt1;

ResultSet rs,rs1,rs2;

public BookingMang() {

initComponents();

}

public final void displayRecord(){

try

{

jTextField1.setText(rs.getString(1));

jTextField2.setText(rs.getString(2));

jTextField3.setText(Integer.toString(rs.getInt(3)));

jTextField4.setText(rs.getString(4));

}

catch(Exception ec)

{}

}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton5ActionPerformed

try {

rs.first();

if (rs.isFirst()) {

rs.first();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton5ActionPerformed

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton7ActionPerformed

try {

rs.next();

if (rs.isLast()) {

rs.last();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton7ActionPerformed

private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton8ActionPerformed

try {

rs.last();

if (rs.isLast()) {

rs.last();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton8ActionPerformed

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton6ActionPerformed

try {

rs.previous();

if (rs.isFirst()) {

rs.first();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton6ActionPerformed

BookRoom.Java

import java.awt.Color;

import java.text.SimpleDateFormat;

import java.util.Date;

import java.sql.\*;

import java.text.DateFormat;

import java.util.Calendar;

import javax.swing.JOptionPane;

public class BookRoom extends javax.swing.JFrame {

Connection con;

Statement smt1,smt2,smt3;

ResultSet rs;

DateFormat formatter;

String roomno,bed,roomt,rate,strdtver1,strdtver2;

String cname,cadd,ccity,cstat,ccoun,cph,cproof;

String cadult,cchild,cmars,cidno,cnation,cpur,cbill;

String diffd;

int amt,bookid;

String dt1,dt2;

/\*\* Creates new form BookRoom \*/

public BookRoom(Object o1,Object o2,Object o3,Object o4,Date o5,Date o6,String diff) {

initComponents();

try{

SimpleDateFormat sdf= new SimpleDateFormat("dd-MMM-yyyy");

SimpleDateFormat sdf1= new SimpleDateFormat("yyyy-MM-dd");

roomno=o1.toString();

bed=o2.toString();

roomt=o3.toString();

rate=o4.toString();

strdtver1=(String) sdf.format(o5.getTime());

strdtver2=(String) sdf.format(o6.getTime());

Calendar c1=Calendar.getInstance();

c1.setTime(o5);

dt1=(String) sdf1.format(c1.getTime());

Calendar c2=Calendar.getInstance();

c2.setTime(o6);

dt2=(String) sdf1.format(c2.getTime());

Class .forName("com.mysql.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/hotel","root","");

smt1=con.createStatement();

diffd=diff;

}

catch(Exception ex)

{

}

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {//GEN-FIRST:event\_formWindowOpened

try{

Color c=new Color(240, 153, 153);

Color c1=new Color(204, 0, 0);

jPanel1.setBackground(c);

jPanel2.setBackground(c1);

String bidsql="select book\_id from bookingtable;";

smt3=con.createStatement();

rs=smt3.executeQuery(bidsql);

int row=0;

int bid=0;

while(rs.next()){

bid=rs.getInt(1);

row++;

}

if(row==0)

{

bid++;

}

else

{

rs.last();

bid=rs.getInt(1);

bid=bid+1;

}

bookid=bid;

jLabel3.setText(roomno);

jLabel6.setText(bed);

jLabel8.setText(roomt);

jLabel10.setText(rate);

jLabel12.setText(strdtver1);

jLabel14.setText(strdtver2);

jTextField13.setText(diffd+" day(s)");

int r=Integer.parseInt(rate);

int day=Integer.parseInt(diffd);

int tax=(r\*day)\*12/100;

amt=(r\*day)+tax;

jTextField15.setText(Integer.toString(amt)+".00");

}

catch(Exception ex)

{

ex.printStackTrace();

}

}//GEN-LAST:event\_formWindowOpened

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton1ActionPerformed

try

{

cname=jTextField1.getText();

cadd=jTextField2.getText();

ccity=jTextField3.getText();

cstat=jTextField4.getText();

ccoun=jTextField5.getText();

cph=jTextField6.getText();

cproof=jTextField9.getText();

cadult=jTextField11.getText();

cchild=jTextField12.getText();

cmars=jTextField14.getText();

cidno=jTextField10.getText();

cnation=jTextField7.getText();

cpur=jTextField8.getText();

int code=JOptionPane.showConfirmDialog(this,"Room No. " + roomno + " is booked for Mr./Ms. " + cname + " for "

+ "diffd..!","Confirmation",JOptionPane.YES\_NO\_CANCEL\_OPTION,JOptionPane.INFORMATION\_MESSAGE);

if(code==JOptionPane.YES\_NO\_OPTION)

{

String insqlcust="insert into custdetail (Cust\_name,Cust\_add,Cust\_city,Cust\_state,Cust\_country,Cust\_ph,Cust\_proof,"

+ "Cust\_adult,Cust\_child,Cust\_mar\_stat,Cust\_id\_no,Cust\_nation,Cust\_purpos,Cust\_bill,book\_id) "

+ "values ('" + cname + "','" + cadd + "','" + ccity + "','" + cstat + "','" + ccoun + "','" + cph + "','" + cproof

+ "','" + cadult + "','" + cchild + "','" + cmars + "','" + cidno +"','" + cnation + "','" + cpur + "'," + amt

+ ",'" + bookid + "');";

smt1=con.createStatement();

int ins=smt1.executeUpdate(insqlcust);

String insqlbook="insert into bookingtable (book\_id,room\_no,date\_fro,date\_to,no\_of\_day) values( '"

+ bookid + "','" + roomno + "','" + dt1 + "','" + dt2 + "','" + diffd + "');";

smt2=con.createStatement();

int ins1=smt1.executeUpdate(insqlbook);

JOptionPane.showMessageDialog( this,"Room Booked...!");

}

else

{

}

}

catch(Exception ex)

{

ex.printStackTrace();

}

}//GEN-LAST:event\_jButton1ActionPerformed

/\*\*

\* @param args the command line arguments

public static void main(String args[]) {

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new BookRoom().setVisible(true);

}

});}

Customer Mang.Java

import java.sql.\*;

import javax.swing.JOptionPane;

public class CustomerMang extends javax.swing.JFrame {

Connection con;

Statement smt,smt1;

ResultSet rs,rs1,rs2;

String ccno,cname,cadd,ccity,cstate,ccount,cph,cproof,cadult,ckid,cmar,cprno,cnation,cpurp,cbamt,cbid;

/\*

1 2 3 4 5 6 7 11 12 9 8 10 13 14 15

Cust\_name,Cust\_add,Cust\_city,Cust\_state,Cust\_country,Cust\_ph,Cust\_proof,Cust\_adult,Cust\_child,Cust\_mar\_stat,Cust\_id\_no,Cust\_nation,Cust\_purpos,Cust\_bill,book\_id

2 3 4 5 6 7 8 9 10 11 12 13 14 15

ccno,cname,cadd,ccity,cstate,ccount,cph,cproof,cadult,ckid,cmar,cprno,cnation,cpurp,cbamt,cbid

\*/

public CustomerMang() {

initComponents();

disableTextF();

try

{

Class.forName("com.mysql.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/hotel","root","");

String query = "SELECT \* FROM custdetail ORDER BY Cust\_No;";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception e)

{

e.printStackTrace();

System.out.println(e.getMessage());

}

}

//FETCH THE VALUES FROM DATABASE TO TEXT FIELDS AND TEXT AREA

public final void displayRecord(){

try

{

jLabel27.setText(Integer.toString(rs.getInt(1)));

jTextField1.setText(rs.getString(2));

jTextField2.setText(rs.getString(3));

jTextField3.setText(rs.getString(4));

jTextField4.setText(rs.getString(5));

jTextField5.setText(rs.getString(6));

jTextField6.setText(rs.getString(7));

jTextField7.setText(rs.getString(8));

jTextField8.setText(rs.getString(12));

jTextField9.setText(rs.getString(11));

jTextField10.setText(rs.getString(13));

jTextField11.setText(rs.getString(9));

jTextField12.setText(rs.getString(10));

jTextField13.setText(rs.getString(14));

jTextField14.setText(Integer.toString(rs.getInt(16)));

jTextField15.setText(Integer.toString(rs.getInt(15)));

}

catch(Exception ec)

{}

}

//GET THE VALUES FROM TEXT FIELDS AND AREA

public void fetchFromTextF(){

cname=jTextField1.getText();

cadd=jTextField2.getText();

ccity=jTextField3.getText();

cstate=jTextField4.getText();

ccount=jTextField5.getText();

cph=jTextField6.getText();

cproof=jTextField7.getText();

cadult=jTextField11.getText();

ckid=jTextField12.getText();

cmar=jTextField9.getText();

cprno=jTextField8.getText();

cnation=jTextField10.getText();

cpurp=jTextField13.getText();

cbid=jTextField14.getText();

cbamt=jTextField15.getText();

}

//DISABLE TEXT FIELDS AND AREA

private void disableTextF(){

jTextField1.setEditable(false);

jTextField2.setEditable(false);

jTextField3.setEditable(false);

jTextField4.setEditable(false);

jTextField5.setEditable(false);

jTextField6.setEditable(false);

jTextField7.setEditable(false);

jTextField8.setEditable(false);

jTextField9.setEditable(false);

jTextField10.setEditable(false);

jTextField11.setEditable(false);

jTextField12.setEditable(false);

jTextField13.setEditable(false);

jTextField14.setEditable(false);

jTextField15.setEditable(false);

jTextField16.setEditable(false);

jTextField17.setEditable(false);

}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton5ActionPerformed

try {

rs.first();

if (rs.isFirst()) {

rs.first();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton5ActionPerformed

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton7ActionPerformed

try {

rs.next();

if (rs.isLast()) {

rs.last();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton7ActionPerformed

private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton8ActionPerformed

try {

rs.last();

if (rs.isLast()) {

rs.last();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton8ActionPerformed

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton6ActionPerformed

try {

rs.previous();

if (rs.isFirst()) {

rs.first();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton6ActionPerformed

Login.Java

import java.sql.\*;

import java.awt.Color;

import javax.swing.ImageIcon;

import javax.swing.JOptionPane;

public class login extends javax.swing.JFrame {

Connection con;

Statement stmt,smt1;

ResultSet rs;

java.util.Date dt1;

//Border bor=new TitledBorder(null,"Check",2,0);

//jPanel1.setBorder(bor);

public login() {

initComponents();

try

{

Class .forName("com.mysql.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/hotel","root","krishna");

}

catch(Exception e)

{

System.out.println(e.getMessage());

}

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton3ActionPerformed

//Point but=new Point(0,250);

//jButton3.setLocation(but);

//Point y=jButton3.getLocation();System.out.println(y);

jButton3.setVisible(false);

//Point pan=new Point(400,250);

//jPanel1.setLocation(pan);

//Point x=jPanel1.getLocation();System.out.println(x);

jPanel1.setVisible(true);

jLabel3.setVisible(true);

jLabel4.setVisible(true);

jTextField1.setVisible(true);

jPasswordField1.setVisible(true);

jButton1.setVisible(true);

jButton2.setVisible(true);

//x=jPanel1.getLocation();System.out.println(x);

}//GEN-LAST:event\_jButton3ActionPerformed

private void formWindowOpened(java.awt.event.WindowEvent evt) {//GEN-FIRST:event\_formWindowOpened

jPanel1.setOpaque(true);

jPanel2.setOpaque(true);

Color c=new Color(240, 153, 153);

Color c1=new Color(170, 119, 98);

jButton3.setBackground(c1);

jLabel3.setVisible(false);

jLabel4.setVisible(false);

jTextField1.setVisible(false);

jPasswordField1.setVisible(false);

jButton1.setVisible(false);

jButton2.setVisible(false);

}//GEN-LAST:event\_formWindowOpened

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton1ActionPerformed

try{

int flag=0;

String sql="Select \* from login;";

smt1=con.createStatement();

rs=smt1.executeQuery(sql);

String log=jTextField1.getText();

String pass =new String(jPasswordField1.getPassword());

while(rs.next())

{

if(log.equals(rs.getString(1)) && pass.equals(rs.getString(2)))

{

flag=1;

break;

}

}

if(flag==1)

{

jLabel3.setEnabled(false);

jLabel4.setEnabled(false);

jTextField1.setEnabled(false);

jPasswordField1.setEnabled(false);

jButton1.setEnabled(false);

jButton2.setEnabled(false);

NavigationFrame nf=new NavigationFrame(this);

nf.setVisible(true);

}

else

{

JOptionPane.showMessageDialog(this, "Error", "Please check user name / password",JOptionPane.ERROR\_MESSAGE);

}

}

catch(Exception ex){

}

}//GEN-LAST:event\_jButton1ActionPerformed

private void jButton3MouseEntered(java.awt.event.MouseEvent evt) {//GEN-FIRST:event\_jButton3MouseEntered

Color c1=new Color(170, 119, 98);

jButton3.setBackground(c1);

}//GEN-LAST:event\_jButton3MouseEntered

private void jButton3MouseExited(java.awt.event.MouseEvent evt) {//GEN-FIRST:event\_jButton3MouseExited

Color c1=new Color(170, 119, 98);

jButton3.setBackground(c1);

}//GEN-LAST:event\_jButton3MouseExited

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton2ActionPerformed

jLabel3.setVisible(false);

jLabel4.setVisible(false);

jTextField1.setVisible(false);

jPasswordField1.setVisible(false);

jButton1.setVisible(false);

jButton2.setVisible(false);

jButton3.setVisible(true);

}//GEN-LAST:event\_jButton2ActionPerformed

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new login().setVisible(true);

}

});

}

NavigationFrame.Java

public class NavigationFrame extends javax.swing.JFrame {

/\*\*

\* Creates new form NavigationFrame

\*/

public NavigationFrame(login O) {

O.setVisible(false);

initComponents();

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {//GEN-FIRST:event\_formWindowOpened

jButton1.setToolTipText("Add/ Edit/Remove Hotel Staff");

jButton2.setToolTipText("Add/ Edit/Remove Hotel Room");

jButton3.setToolTipText("Check Room is Available or Not");

jButton4.setToolTipText("Close the application");

}//GEN-LAST:event\_formWindowOpened

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton1ActionPerformed

StaffMang sm=new StaffMang();

sm.setVisible(true);

}//GEN-LAST:event\_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton2ActionPerformed

RoomMang rm=new RoomMang();

rm.setVisible(true);

}//GEN-LAST:event\_jButton2ActionPerformed

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton3ActionPerformed

Availability avail=new Availability();

avail.setVisible(true);

}//GEN-LAST:event\_jButton3ActionPerformed

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton5ActionPerformed

CustomerMang cm=new CustomerMang();

cm.setVisible(true);

}//GEN-LAST:event\_jButton5ActionPerformed

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton4ActionPerformed

System.exit(0);

}//GEN-LAST:event\_jButton4ActionPerformed

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton6ActionPerformed

BookingMang bm=new BookingMang();

bm.setVisible(true);

}//GEN-LAST:event\_jButton6ActionPerformed

RoomMang.Java

import java.sql.\*;

import javax.swing.JOptionPane;

public class RoomMang extends javax.swing.JFrame {

Connection con;

Statement smt,smt1;

ResultSet rs,rs1,rs2;

String rno,rtype,rbed;

int rrate;

public RoomMang() {

initComponents();

try

{

Class.forName("com.mysql.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/hotel","root","");

String query = "SELECT \* FROM roomdetail ORDER BY room\_no;";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception e)

{

e.printStackTrace();

System.out.println(e.getMessage());

}

}

//FETCH THE VALUES FROM DATABASE TO TEXT FIELDS AND TEXT AREA

public final void displayRecord(){

try

{

jTextField1.setText(rs.getString(1));

jTextField2.setText(rs.getString(2));

jTextField3.setText(Integer.toString(rs.getInt(3)));

jTextField4.setText(rs.getString(4));

}

catch(Exception ec)

{}

}

//GET THE VALUES FROM TEXT FIELDS AND AREA

public void fetchFromTextF(){

rno=jTextField1.getText();

rtype=jTextField2.getText();

rrate=Integer.parseInt(jTextField3.getText());

rbed=jTextField4.getText();

}

//DISABLE TEXT FIELDS AND AREA

public void disableTextF(){

jTextField1.setEditable(false);

jTextField2.setEditable(false);

jTextField3.setEditable(false);

jTextField4.setEditable(false);

}

//CLEAR TEXT FIELDS AND AREA

public void clearTextF(){

try

{

jTextField1.setText(null);

jTextField2.setText(null);

jTextField3.setText(null);

jTextField4.setText(null);

}

catch(Exception ec)

{}

}

//ENABLE TEXT FIELDS AND AREA

public void enableTextF(){

jTextField1.setEditable(true);

jTextField2.setEditable(true);

jTextField3.setEditable(true);

jTextField4.setEditable(true);

}

//MOVE ON FIRST ROOM

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton5ActionPerformed

try {

rs.first();

if (rs.isFirst()) {

rs.first();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton5ActionPerformed

//MOVE ON NEXT ROOM

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton7ActionPerformed

try {

rs.next();

if (rs.isLast()) {

rs.last();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton7ActionPerformed

//MOVE ON LAST ROOM

private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton8ActionPerformed

try {

rs.last();

if (rs.isLast()) {

rs.last();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton8ActionPerformed

//MOVE ON PREVIOUS ROOM

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton6ActionPerformed

try {

rs.previous();

if (rs.isFirst()) {

rs.first();

}

displayRecord();

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton6ActionPerformed

//ADD NEW ROOM

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton1ActionPerformed

try {

clearTextF();

enableTextF();

String query = "SELECT \* FROM roomdetail where room\_no like '" + jTextField1.getText() + "';";

smt = con.createStatement();

rs1 = smt.executeQuery(query);

if (!rs1.next()) {

try {

if (evt.getActionCommand().equals("Save")) {

fetchFromTextF();

int code = JOptionPane.showConfirmDialog(this, "Information of Room No." + rno + " will be added in database.", "Confirmation", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.INFORMATION\_MESSAGE);

if (code == JOptionPane.YES\_OPTION) {

String insertquery = "insert into roomdetail(room\_no,room\_type,room\_rate,room\_bed)values('" + rno + "','" + rtype + "'," + rrate + ",'" + rbed + "');";

smt = con.createStatement();

int success = smt.executeUpdate(insertquery);

if (success > 0) {

JOptionPane.showMessageDialog(this, "Record Saved");

jButton1.setText("New");

} else {

JOptionPane.showMessageDialog(this, "Problem in Saving. Retry");

}

} else {

}

} else if (evt.getActionCommand().equals("New")) {

clearTextF();

jButton1.setText("Save");

}

} catch (Exception ex) {

ex.printStackTrace();

}

} else {

JOptionPane.showMessageDialog(this, "Room No. already used, Give another Room No.");

}

} catch (Exception ex) {

}

}//GEN-LAST:event\_jButton1ActionPerformed

//EDIT AND UPDATE EXISTING ROOM

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton2ActionPerformed

enableTextF();

jTextField1.setEnabled(false);

try {

if (evt.getActionCommand().equals("Update")) {

fetchFromTextF();

int code = JOptionPane.showConfirmDialog(this, "Information of Room No." + rno + " will change in database.", "Confirmation", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.INFORMATION\_MESSAGE);

if (code == JOptionPane.YES\_OPTION) {

String updatequery = "update roomdetail set room\_no = '" + rno + "',room\_type='" + rtype + "',room\_rate=" + rrate + ",room\_bed='" + rbed + "';";

smt = con.createStatement();

int success = smt.executeUpdate(updatequery);

if (success > 0) {

JOptionPane.showMessageDialog(this, "Record Modified");

jButton1.setText("Edit");

} else {

JOptionPane.showMessageDialog(this, "Problem in mofifying. Retry");

}

} else {

}

} else if (evt.getActionCommand().equals("Edit")) {

JOptionPane.showMessageDialog(this, "You are about to change the existing information of current selected room");

jButton2.setText("Update");

}

} catch (Exception ex) {

ex.printStackTrace();

}

}//GEN-LAST:event\_jButton2ActionPerformed

//DELETE EXISTING ROOM

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton3ActionPerformed

try {

if (evt.getActionCommand().equals("Confirm")) {

fetchFromTextF();

int code = JOptionPane.showConfirmDialog(this, "Room No" + rno + " will be deleted from database.", "Confirmation", JOptionPane.YES\_NO\_CANCEL\_OPTION, JOptionPane.INFORMATION\_MESSAGE);

if (code == JOptionPane.YES\_OPTION) {

String updatequery = "DELETE from roomdetail where room\_no like '" + rno + "';";

smt = con.createStatement();

int success = smt.executeUpdate(updatequery);

if (success > 0) {

JOptionPane.showMessageDialog(this, "Record Deleted");

jButton1.setText("Delete");

} else {

JOptionPane.showMessageDialog(this, "Problem in deletion. Retry");

}

} else {

}

} else if (evt.getActionCommand().equals("Delete")) {

JOptionPane.showMessageDialog(this, "Your confirmation will delete the record of current selected room");

jButton2.setText("Confirm");

}

} catch (Exception ex) {

ex.printStackTrace();

}

}//GEN-LAST:event\_jButton3ActionPerformed

//SEARCH ROOM NO

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton4ActionPerformed

disableTextF();

String srno=jTextField5.getText();

if(rno.equals(""))

{

JOptionPane.showMessageDialog(this, "Enter Room No. you want to search");

}

else if(!rno.equals(""))

{

try{

String query = "SELECT \* FROM roomdetail where room\_no LIKE '"+srno+"';";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

}//GEN-LAST:event\_jButton4ActionPerformed

StaffMang.Java

import java.sql.\*;

import javax.swing.JOptionPane;

public class StaffMang extends javax.swing.JFrame {

Connection con;

Statement smt,smt1;

ResultSet rs,rs1,rs2;

String sid,spass,sfname,slname,sdesig,sadd,snum;

/\*CONSTRUCTOR OF CLASS WHERE DATABASE CONNECTION IS MADE AND

OBJECT OF RESULTSET AND STATEMENT IS CREATED\*/

public StaffMang() {

initComponents();

try

{

Class.forName("com.mysql.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/hotel","root","");

String query = "SELECT \* FROM login ORDER BY userid;";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception e)

{

e.printStackTrace();

System.out.println(e.getMessage());

}

}

public final void displayRecord(){

try

{

jTextField1.setText(rs.getString(1));

jTextField2.setText(rs.getString(2));

jTextField4.setText(rs.getString(3));

jTextField5.setText(rs.getString(4));

jTextField6.setText(rs.getString(5));

jTextField7.setText(rs.getString(7));

jTextArea1.setText(rs.getString(6));

}

catch(Exception ec)

{}

}

//ENABLE TEXT FIELDS AND AREA

public void enableTextF(){

jTextField1.setEditable(true);

jTextField2.setEditable(true);

jTextField3.setEditable(true);

jTextField4.setEditable(true);

jTextField5.setEditable(true);

jTextField6.setEditable(true);

jTextArea1.setEditable(true);

}

//DISABLE TEXT FIELDS AND AREA

public void disableTextF(){

jTextField1.setEditable(false);

jTextField2.setEditable(false);

jTextField3.setEditable(false);

jTextField4.setEditable(false);

jTextField5.setEditable(false);

jTextField6.setEditable(false);

jTextArea1.setEditable(false);

}

//CLEAR TEXT FIELDS AND AREA

public void clearTextF(){

try

{

jTextField1.setText(null);

jTextField2.setText(null);

jTextField4.setText(null);

jTextField5.setText(null);

jTextField6.setText(null);

jTextField7.setText(null);

jTextArea1.setText(null);

}

catch(Exception ec)

{}

}

//GET THE VALUES FROM TEXT FIELDS AND AREA

public void fetchFromTextF(){

sid=jTextField1.getText();

spass=jTextField2.getText();

if(!spass.equals(jTextField3.getText()))

{

JOptionPane.showMessageDialog(this, "Re-entered password is not same");

}

sdesig=jTextField4.getText();

sfname=jTextField5.getText();

slname=jTextField6.getText();

snum=jTextField7.getText();

sadd=jTextArea1.getText();

}

//EDIT STAFF RECORD

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton2ActionPerformed

enableTextF();

jTextField1.setEnabled(false);

jTextField3.setText(spass);

try{

if(evt.getActionCommand().equals("Update"))

{

fetchFromTextF();

int code=JOptionPane.showConfirmDialog(this,"Information of " + sfname + " will change in database.","Confirmation",JOptionPane.YES\_NO\_CANCEL\_OPTION,JOptionPane.INFORMATION\_MESSAGE);

if(code==JOptionPane.YES\_OPTION)

{

String updatequery="update login set userid = '"+sid+"',password='"+spass+"',Staffdesig='"+sdesig+"',StaffFName='"+sfname+"',StaffLName='"+slname+"',StaffAdd='"+sadd+"',StaffContactNo='"+snum+"' where userid like '"+sid+"';";

smt=con.createStatement();

int success=smt.executeUpdate(updatequery);

if(success>0)

{

JOptionPane.showMessageDialog(this, "Record Modified");

jButton1.setText("Edit");

}

else

{

JOptionPane.showMessageDialog(this, "Problem in mofifying. Retry");

}

}

else

{ }

}

else if(evt.getActionCommand().equals("Edit"))

{

JOptionPane.showMessageDialog(this, "You are about to change the existing information of current selected staff");

jButton2.setText("Update");

}

}

catch(Exception ex)

{

ex.printStackTrace();

}

}//GEN-LAST:event\_jButton2ActionPerformed

//SEARCH STAFF RECORD

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton4ActionPerformed

disableTextF();

String seaID=jTextField8.getText();

String seaFN=jTextField9.getText();

String seaLN=jTextField10.getText();

if(seaID.equals("")&&seaFN.equals("")&&seaLN.equals(""))

{

JOptionPane.showMessageDialog(this, "Select at least 1 option for search");

}

else

{

if((!seaID.equals("")) && (!seaFN.equals("")) && (!seaLN.equals("")))

{

try{

String query = "SELECT \* FROM login where userid LIKE '"+seaID+"' AND StaffFName LIKE '"+seaFN+"' AND StaffLName LIKE '"+seaLN+"';";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

else if((!seaID.equals("")) && (!seaFN.equals("")) && seaLN.equals(""))

{

try{

String query = "SELECT \* FROM login where userid LIKE '"+seaID+"' AND StaffFName LIKE '"+seaFN+"';";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

else if((!seaID.equals("")) && seaFN.equals("") && (!seaLN.equals("")))

{

try{

String query = "SELECT \* FROM login where userid LIKE '"+seaID+"' AND StaffLName LIKE '"+seaLN+"';";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

else if(seaID.equals("") && (!seaFN.equals("")) && (!seaLN.equals("")))

{

try{

String query = "SELECT \* FROM login where StaffFName LIKE '"+seaFN+"' AND StaffLName LIKE '"+seaLN+"';";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

else if((!seaID.equals("")) && seaFN.equals("") && seaLN.equals(""))

{

try{

String query = "SELECT \* FROM login where userid LIKE '"+seaID+"';";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

else if(seaID.equals("") && (!seaFN.equals("")) && seaLN.equals(""))

{

try{

String query = "SELECT \* FROM login where StaffFName LIKE '"+seaFN+"';";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

else if(seaID.equals("") && seaFN.equals("") && (!seaLN.equals("")))

{

try{

String query = "SELECT \* FROM login where StaffLName LIKE '"+seaLN+"';";

smt=con.createStatement();

rs=smt.executeQuery(query);

rs.next();

displayRecord();

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

}

}//GEN-LAST:event\_jButton4ActionPerformed

//MOVE ON LAST RECORD

private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton8ActionPerformed

try

{

rs.last();

if(rs.isLast())

rs.last();

displayRecord();

}

catch(Exception ex) {}

}//GEN-LAST:event\_jButton8ActionPerformed

//MOVE ON PREVIOUS RECORD

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton6ActionPerformed

try

{

rs.previous();

if(rs.isFirst())

rs.first();

displayRecord();

}

catch(Exception ex) {}

}//GEN-LAST:event\_jButton6ActionPerformed

//MOVE ON FIRST RECORD

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton5ActionPerformed

try

{

rs.first();

if(rs.isFirst())

rs.first();

displayRecord();

}

catch(Exception ex) {}

}//GEN-LAST:event\_jButton5ActionPerformed

//MOVE ON NEXT RECORD

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton7ActionPerformed

try

{

rs.next();

if(rs.isLast())

rs.last();

displayRecord();

}

catch(Exception ex) {}

}//GEN-LAST:event\_jButton7ActionPerformed

//TOOL TIPS SETTING

private void formWindowOpened(java.awt.event.WindowEvent evt) {//GEN-FIRST:event\_formWindowOpened

jTextField1.setToolTipText("This is an Unique ID");

jTextField3.setToolTipText("Re Enter the same Password");

jButton4.setToolTipText("First select the option for searching");

jTextField8.setToolTipText("User ID is case sensitive");

jTextField9.setToolTipText("First Name is case sensitive");

jTextField10.setToolTipText("Last Name is case sensitive");

jButton3.setToolTipText("Current Selected Staff Record will be DELETED");

jButton9.setToolTipText("Close Window");

}//GEN-LAST:event\_formWindowOpened

//ADD RECORD OF STAFF

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton1ActionPerformed

try{

clearTextF();

enableTextF();

String query = "SELECT \* FROM login where userid like '"+jTextField1.getText()+"';";

smt=con.createStatement();

rs1=smt.executeQuery(query);

if(!rs1.next()){

try{

if(evt.getActionCommand().equals("Save"))

{

fetchFromTextF();

int code=JOptionPane.showConfirmDialog(this,"Information of " + sfname + " will be added in database.","Confirmation",JOptionPane.YES\_NO\_CANCEL\_OPTION,JOptionPane.INFORMATION\_MESSAGE);

if(code==JOptionPane.YES\_OPTION)

{

String insertquery="insert into login(userid,password,Staffdesig,StaffFName,StaffLName,StaffAdd,StaffContactNo)values('"+sid+"','"+spass+"','"+sdesig+"','"+sfname+"','"+slname+"','"+sadd+"','"+snum+"');";

smt=con.createStatement();

int success=smt.executeUpdate(insertquery);

if(success>0)

{

JOptionPane.showMessageDialog(this, "Record Saved");

jButton1.setText("New");

}

else

{

JOptionPane.showMessageDialog(this, "Problem in Saving. Retry");

}

}

else

{ }

}

else if(evt.getActionCommand().equals("New"))

{

clearTextF();

jButton1.setText("Save");

}

}

catch(Exception ex)

{

ex.printStackTrace();

}

}

else

JOptionPane.showMessageDialog(this,"User ID already taken, Give another User ID");

}

catch(Exception ex)

{ }

}//GEN-LAST:event\_jButton1ActionPerformed

//DELETE RECORD OF STAFF

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton3ActionPerformed

jTextField3.setText(spass);

try{

if(evt.getActionCommand().equals("Confirm"))

{

fetchFromTextF();

int code=JOptionPane.showConfirmDialog(this,"Information of " + sfname + " will be deleted from database.","Confirmation",JOptionPane.YES\_NO\_CANCEL\_OPTION,JOptionPane.INFORMATION\_MESSAGE);

if(code==JOptionPane.YES\_OPTION)

{

String updatequery="DELETE from login where userid like '"+sid+"';";

smt=con.createStatement();

int success=smt.executeUpdate(updatequery);

if(success>0)

{

JOptionPane.showMessageDialog(this, "Record Deleted");

jButton1.setText("Delete");

}

else

{

JOptionPane.showMessageDialog(this, "Problem in deletion. Retry");

}

}

else

{ }

}

else if(evt.getActionCommand().equals("Delete"))

{

JOptionPane.showMessageDialog(this, "Your confirmation will delete the record of current selected staff");

jButton2.setText("Confirm");

}

}

catch(Exception ex)

{

ex.printStackTrace();

}

}//GEN-LAST:event\_jButton3ActionPerformed

//CLOSE BUTTON

private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_jButton9ActionPerformed

this.dispose();

}//GEN-LAST:event\_jButton9ActionPerformed