Appendix I - Code

```
import java.sql.*;
import javax.swing.*;
import java.lang.*;

public class database {

//this class connects to the server of the database and makes it possible for the
statements to be executed

    Connection conn = null;
    public static Connection ConnecrDb() throws SQLException {
        try{
           Class.forName("org.sqlite.JDBC");

        Connection conn = DriverManager.getConnection("jdbc:sqlite:
mydatabase.sqlite");

        return conn;
    }
    catch (ClassNotFoundException | SQLException e){
    JOptionPane.showMessageDialog(null, e);
}

        return null;
    }
}
```

Appendix I - Code

```
import java.sql.*;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.*;

public class LogIn extends javax.swing.JFrame {

    public LogIn() {
           initComponents();
    }

    private void initComponents() {
//elements are initiated automatically by NetBeans
                  }

    private void signUpMouseClicked(java.awt.event.MouseEvent evt) {
createAccount a = new createAccount(); //this method opens the Create Account frame
when user clicks on the signup button
a.setVisible(true);
this.setVisible(false);
    }

    private void signUpActionPerformed(java.awt.event.ActionEvent evt) {

    }

    private void logInActionPerformed(java.awt.event.ActionEvent evt) {
  String a = (String)statusSelect.getSelectedItem();
        if (username.getText().trim().isEmpty() &&
password.getText().trim().isEmpty() && a.equals("Select")){
      jLabel6.setText("No status selected.");
      jLabel7.setText("Username is empty.");
      jLabel8.setText("Password is empty.");
  }//this shows an error message if any of the fields are left empty
        else if (username.getText().trim().isEmpty() &&
password.getText().trim().isEmpty() ){
      jLabel7.setText("Username is empty.");
      jLabel8.setText("Password is empty.");
        }
        else if (password.getText().trim().isEmpty() && a.equals("Select")){
      jLabel6.setText("No status selected.");
      jLabel8.setText("Password is empty.");
        }
        else if (username.getText().trim().isEmpty() && a.equals("Select")){
      jLabel6.setText("No status selected.");
      jLabel7.setText("Username is empty.");
        }
        else if(username.getText().trim().isEmpty()){
            jLabel7.setText("Username is empty.");
        }
        else if(password.getText().trim().isEmpty()){
```

```
            jLabel8.setText("Password is empty.");
        }
        else if(a.equals("Select")){
            jLabel6.setText("No status selected.");
        }

  String un, pass;

        un = username.getText();
        pass = password.getText();
        try {
            Class.forName("org.sqlite.JDBC");
        } catch (ClassNotFoundException ex) {
            Logger.getLogger(CreateOpportunity.class.getName()).log(Level.SEVERE,
null, ex);
        }//connecting to the server

        Connection conn = null;
        try{ //this part of the method connects to the database through a sql
statemenent and executes it

   String url =
"jdbc:sqlite:C:\\Users\\Mario\\Documents\\NetBeansProjects\\IA\\mydatabase.sqlite";
             conn = DriverManager.getConnection(url,"","");
            Statement st = conn.createStatement();

           String sql = ("SELECT*FROM Info WHERE Username='"+un+"' AND
Password='"+pass+"'AND Status='"+a+"'");
           PreparedStatement stat = conn.prepareStatement(sql);
             ResultSet rs = stat.executeQuery();
            conn.close();
if (rs.next()) { //if this statement is executed, the user logins

        Browse br = new Browse();
br.setVisible(true);
this.setVisible(false);}
else if ((!username.getText().trim().isEmpty() &&
!password.getText().trim().isEmpty() && !a.equals("Select"))||
        (!username.getText().trim().isEmpty() &&
!password.getText().trim().isEmpty())){
    JOptionPane.showMessageDialog(null, "Username or Password are incorrect!");
    //if the user imputs non-matchig password and username an error message appears
}

        }
        catch(Exception e){
        JOptionPane.showMessageDialog(null, e);
        }

    }
```

```
    private void statusSelectMouseClicked(java.awt.event.MouseEvent evt) {
        jLabel6.setText("");
    }

    private void usernameKeyReleased(java.awt.event.KeyEvent evt) {
        jLabel7.setText("");
    }

    private void passwordKeyReleased(java.awt.event.KeyEvent evt) {
        jLabel8.setText("");
    }


    public static void main(String args[]) {
… //generated code by NetBeans
        java.awt.EventQueue.invokeLater(new Runnable() {
            @Override
            public void run() {
                new LogIn().setVisible(true);

            }
        });
    }
}
```

```
import java.awt.HeadlessException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.JTextField;


public class createAccount extends javax.swing.JFrame {

    public createAccount() {
        initComponents();
    }

    private void initComponents() {
//elements are initiated automatically by NetBeans
    }

    private void cancelActionPerformed(java.awt.event.ActionEvent evt) {
LogIn login = new LogIn();
login.setVisible(true);
this.setVisible(false);
    }

    private void submitActionPerformed(java.awt.event.ActionEvent evt) {
 String fname, lname, email, username, password, st, bd, qst, ans;
            fname = FName.getText();
            lname = setLName.getText();
            email = setEmail.getText();
            username = setUsername.getText();
            password = setPass.getText();
            st = (String)setStatus.getSelectedItem();
            bd =
((JTextField)jDateChooser1.getDateEditor().getUiComponent()).getText();
            qst = (String)question.getSelectedItem();
            ans = answer.getText();



 if
(fname.isEmpty()||lname.isEmpty()||email.isEmpty()||username.isEmpty()||password.equa
ls("Select")||st.isEmpty()||bd.isEmpty()
         ||qst.equals("Select")||ans.isEmpty()){
     JOptionPane.showMessageDialog(null, "All fields are required!");
```

```
 }
 else if(!password.equals(confPass.getText())){
     jLabel11.setText("Passwords don't match");
 }

 else{

            try {
            Class.forName("org.sqlite.JDBC");
        } catch (ClassNotFoundException ex) {
            Logger.getLogger(CreateOpportunity.class.getName()).log(Level.SEVERE,
null, ex);
        }
            Connection conn = null;
try{

            String url =
"jdbc:sqlite:C:\\Users\\Mario\\Documents\\NetBeansProjects\\IA\\mydatabase.sqlite";
             conn = DriverManager.getConnection(url);

            String sql = ("INSERT INTO Info (FirstName, LastName, Email, Username,
Password, Status, Birthday, Question, Answer) VALUES ('"+fname+"', '"+lname+"',
'"+email+"', '"+username+"', '"+password+"', '"+st+"', '"+bd+"', '"+qst+"',
'"+ans+"')");

                  PreparedStatement preparedStmt = conn.prepareStatement(sql);
                    preparedStmt.execute();

        JOptionPane.showMessageDialog(null, "New Account created!");
        new LogIn().setVisible(true);
        this.setVisible(false);
        preparedStmt.close();
        }
        catch (HeadlessException | SQLException e){
            JOptionPane.showMessageDialog(null, e);
        }
finally
    {
      try
      {
        if(conn != null)
          conn.close();
      }
      catch(SQLException e)
      {

        System.err.println(e);
      }
    }
 }
```

```
    }
    private void confPassActionPerformed(java.awt.event.ActionEvent evt) {
        if (!confPass.getText().equals(setPass.getText())){
              JOptionPane.showMessageDialog(null, "Password not confirmed!");
        }
    }

    private void confPassKeyReleased(java.awt.event.KeyEvent evt) {
jLabel11.setText("");
    }

    public void main(String args[]) throws SQLException {

        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new createAccount().setVisible(true);
            }
        });

    }
}
```

```
import java.awt.HeadlessException;
import java.sql.*;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import net.proteanit.sql.DbUtils;
import javax.swing.table.TableModel;
import javax.swing.table.TableRowSorter;

public class Browse extends javax.swing.JFrame {

    Connection conn = null;
    ResultSet rs = null;
    PreparedStatement pst = null;

    public Browse() {
        initComponents();

        try {
            conn = database.ConnecrDb();
        } catch (SQLException ex) {
            Logger.getLogger(Browse.class.getName()).log(Level.SEVERE, null, ex);
        }
        updateTable();

    }

    private void updateTable(){
        try{
        String sql = "SELECT * FROM Opportunities";
        pst = conn.prepareStatement(sql);
        rs = pst.executeQuery();
        oppdatabase.setModel(DbUtils.resultSetToTableModel(rs));
        }
        catch (Exception e){
            JOptionPane.showMessageDialog(null, e);
        }
          finally
    {
      try
      {
          rs.close();
          pst.close();
        if(conn != null)
          conn.close();
      }
      catch(SQLException e)
      {

        System.err.println(e);
```

```
}
    }

    }

    private void initComponents() {

    }

    private void logout1ActionPerformed(java.awt.event.ActionEvent evt) {
        new LogIn().setVisible(true);
        this.setVisible(false);
    }

    private void myopportunities1ActionPerformed(java.awt.event.ActionEvent evt) {
        new MyOpportunities().setVisible(true);
        this.setVisible(false);
    }

    private void jScrollPane3MouseClicked(java.awt.event.MouseEvent evt) {

    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        new CreateOpportunity().setVisible(true);

    }


    private void oppdatabaseMouseClicked(java.awt.event.MouseEvent evt) {

    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
         int index = oppdatabase.getSelectedRow();
        TableModel model = oppdatabase.getModel();

        String title = model.getValueAt(index, 0).toString();
        String type = model.getValueAt(index, 1).toString();
        String location = model.getValueAt(index, 3).toString();
        String date1 = model.getValueAt(index, 4).toString();
        String date2 = model.getValueAt(index, 5).toString();
        String description = model.getValueAt(index, 2).toString();
        String requirements = model.getValueAt(index, 6).toString();
        String website = model.getValueAt(index, 7).toString();
        String contact = model.getValueAt(index, 8).toString();
        String applink = model.getValueAt(index, 9).toString();
        String appdeadline = model.getValueAt(index, 10).toString();

            try {
            Class.forName("org.sqlite.JDBC");
```

```
        } catch (ClassNotFoundException ex) {
            Logger.getLogger(CreateOpportunity.class.getName()).log(Level.SEVERE,
null, ex);
        }
            Connection conn = null;
try{

            String url = "jdbc:sqlite:mydatabase.sqlite";

             conn = DriverManager.getConnection(url);

            String sql = ("INSERT INTO PersonalOpp (Title, Type, Description,
Location, Date1, Date2, Requirements, Website, Contact, ApplicationLink,
ApplicationDeadline) "
                    + "VALUES ('"+title+"', '"+type+"', '"+description+"',
'"+location+"', '"+date1+"', '"+date2+"', '"+requirements+"', "
                    + "'"+website+"', '"+contact+"', '"+applink+"',
'"+appdeadline+"')");

                  PreparedStatement preparedStmt = conn.prepareStatement(sql);
                    preparedStmt.execute();


        preparedStmt.close();
        }
        catch (HeadlessException | SQLException e){
            JOptionPane.showMessageDialog(null, e);
        }
finally
    {
      try
      {
        if(conn != null)
          conn.close();
      }
      catch(SQLException e)
      {

        System.err.println(e);
      }
    }

    }

    private void jComboBox1ActionPerformed(java.awt.event.ActionEvent evt) {
String query = jComboBox1.getSelectedItem().toString();

    }

    private void jComboBox1ItemStateChanged(java.awt.event.ItemEvent evt) {
```

```
    }

    private void jTextField1KeyReleased(java.awt.event.KeyEvent evt) {
        String query = jTextField1.getText().toLowerCase();

    }

    private void contacts1ActionPerformed(java.awt.event.ActionEvent evt) {
        new Contacts().setVisible(true);
        this.setVisible(false);
    }

    showOpp a = new showOpp();
    private void oppdatabaseKeyPressed(java.awt.event.KeyEvent evt) {
        int index = oppdatabase.getSelectedRow();
        TableModel model = oppdatabase.getModel();

        String title = model.getValueAt(index, 0).toString();
        String type = model.getValueAt(index, 1).toString();
        String location = model.getValueAt(index, 3).toString();
        String date1 = model.getValueAt(index, 4).toString();
        String date2 = model.getValueAt(index, 5).toString();
        String description = model.getValueAt(index, 2).toString();
        String requirements = model.getValueAt(index, 6).toString();
        String website = model.getValueAt(index, 7).toString();
        String contact = model.getValueAt(index, 8).toString();
        String applink = model.getValueAt(index, 9).toString();
        String appdeadline = model.getValueAt(index, 10).toString();

        a.setVisible(true);
        a.setLocationRelativeTo(null);

        a.title1.setText(title);
        a.type1.setText(type);
        a.location1.setText(location);
        a.stdate1.setText(date1);
        a.enddate1.setText(date2);
        a.description1.setText(description);
        a.requirements1.setText(requirements);
        a.website1.setText(website);
        a.contact1.setText(contact);
        a.applink1.setText(applink);
        a.appdl1.setText(appdeadline);
    }

    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Browse().setVisible(true);
            }
        });} }
```

```
import java.awt.HeadlessException;
import java.sql.Connection;
import java.sql.*;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.JTextField;

    public CreateOpportunity() {
        initComponents();
    }

    private void initComponents() {

    }

    private void cancelActionPerformed(java.awt.event.ActionEvent evt) {
        this.setVisible(false);
    }

    private void createActionPerformed(java.awt.event.ActionEvent evt) {
        String t, tp, loc, desc, req, web, cont, applink;
        t = title.getText();
        tp = (String)type.getSelectedItem();
        loc = location.getText();
        desc = description.getText();
        req = requirements.getText();
        web = website.getText();
        cont = contact.getText();
        applink = applicationlink.getText();

        String dt1, dt2, appdeadline;
        dt1 = ((JTextField)date1.getDateEditor().getUiComponent()).getText();
        dt2 = ((JTextField)date2.getDateEditor().getUiComponent()).getText();
        appdeadline =
((JTextField)deadline.getDateEditor().getUiComponent()).getText();


if(t.isEmpty()||tp.equals("Select")||loc.isEmpty()||desc.isEmpty()||req.isEmpty()||we
b.isEmpty()||

cont.isEmpty()||applink.isEmpty()||dt1.isEmpty()||dt2.isEmpty()||appdeadline.isEmpty(
)){
        JOptionPane.showMessageDialog(null, "All fields are required!");
    }
        else{

        try {
            Class.forName("org.sqlite.JDBC");
        } catch (ClassNotFoundException ex) {
```

```
            Logger.getLogger(CreateOpportunity.class.getName()).log(Level.SEVERE,
null, ex);
        }

    Connection conn = null;
        try{

            String url =
"jdbc:sqlite:C:\\Users\\Mario\\Documents\\NetBeansProjects\\IA\\mydatabase.sqlite";
             conn = DriverManager.getConnection(url,"","");
            Statement st = conn.createStatement();

            String sql1 = ("INSERT INTO Opportunities (Title, Type, Description,
Location, Date1, Date2, Requirements, Website, Contact, ApplicationLink,
ApplicationDeadline) VALUES ('"+t+"', '"+tp+"', '"+loc+"', '"+desc+"', '"+dt1+"',
'"+dt2+"', '"+req+"', '"+web+"', '"+cont+"', '"+applink+"','"+appdeadline+"')");
        PreparedStatement preparedStmt1 = conn.prepareStatement(sql1);
                    preparedStmt1.executeUpdate();
    this.setVisible(false);
    JOptionPane.showMessageDialog(null, "New Opportunity added!");

        st.close();
        preparedStmt1.close();
        }
        catch(HeadlessException | SQLException e){
            JOptionPane.showMessageDialog(null, e);
        }
        finally
    {
      try
      {
        if(conn != null)
          conn.close();
      }
      catch(SQLException e)
      {

        System.err.println(e);
      }
    }
        }
    }


    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new CreateOpportunity().setVisible(true);
            }
        });
    }}
```

```
import java.awt.HeadlessException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;

public class showOpp extends javax.swing.JFrame {

        Connection conn = null;

    public showOpp() {
        initComponents();
        try {
            conn = database.ConnecrDb();
        } catch (SQLException ex) {
            Logger.getLogger(Browse.class.getName()).log(Level.SEVERE, null, ex);
        }
    }

    private void initComponents() {

    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        this.setVisible(false);
    }

    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new showOpp().setVisible(true);
            }
        });
    }
}
```

```
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.table.TableModel;
import net.proteanit.sql.DbUtils;

public class MyOpportunities extends javax.swing.JFrame {

    Connection conn = null;
    ResultSet rs = null;
    PreparedStatement pst = null;

    public MyOpportunities() {
        initComponents();
         try {
            conn = database.ConnecrDb();
        } catch (SQLException ex) {
            Logger.getLogger(Browse.class.getName()).log(Level.SEVERE, null, ex);
        }
        updateTable();
    }
    private void updateTable(){
        try{
        String sql = "SELECT * FROM PersonalOpp";
        pst = conn.prepareStatement(sql);
        rs = pst.executeQuery();
        saved.setModel(DbUtils.resultSetToTableModel(rs));
        }
        catch (Exception e){
            JOptionPane.showMessageDialog(null, e);
        }
          finally
    {
      try
      {
          rs.close();
          pst.close();
      if(conn != null)
          conn.close();
      }
      catch(SQLException e)
      {

        System.err.println(e);
      }
    }
```

```
    }

    private void initComponents() {

    }

    private void browse3ActionPerformed(java.awt.event.ActionEvent evt) {
        new Browse().setVisible(true);
        this.setVisible(false);
    }

    private void logout3ActionPerformed(java.awt.event.ActionEvent evt) {
        new LogIn().setVisible(true);
        this.setVisible(false);
    }

    private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
         new Contacts().setVisible(true);
        this.setVisible(false);
    }
    showOpp a = new showOpp();
    private void savedKeyPressed(java.awt.event.KeyEvent evt) {
        int index = saved.getSelectedRow();
        TableModel model = saved.getModel();

        String title = model.getValueAt(index, 0).toString();
        String type = model.getValueAt(index, 1).toString();
        String location = model.getValueAt(index, 3).toString();
        String date1 = model.getValueAt(index, 4).toString();
        String date2 = model.getValueAt(index, 5).toString();
        String description = model.getValueAt(index, 2).toString();
        String requirements = model.getValueAt(index, 6).toString();
        String website = model.getValueAt(index, 7).toString();
        String contact = model.getValueAt(index, 8).toString();
        String applink = model.getValueAt(index, 9).toString();
        String appdeadline = model.getValueAt(index, 10).toString();

        a.setVisible(true);
        a.setLocationRelativeTo(null);

        a.title1.setText(title);
        a.type1.setText(type);
        a.location1.setText(location);
        a.stdate1.setText(date1);
        a.enddate1.setText(date2);
        a.description1.setText(description);
        a.requirements1.setText(requirements);
        a.website1.setText(website);
        a.contact1.setText(contact);
        a.applink1.setText(applink);
        a.appdl1.setText(appdeadline);
```

```
    }

    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new MyOpportunities().setVisible(true);
            }
        });
    }
}
```

```
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import net.proteanit.sql.DbUtils;

public class Contacts extends javax.swing.JFrame {

    Connection conn = null;
    ResultSet rs = null;
    PreparedStatement pst = null;

    public Contacts() {
        initComponents();
        try {
            conn = database.ConnecrDb();
        } catch (SQLException ex) {
            Logger.getLogger(Browse.class.getName()).log(Level.SEVERE, null, ex);
        }
        updateTable1();
    }

    void updateTable1(){
        try{
        String sql = "SELECT * FROM Contacts";
        pst = conn.prepareStatement(sql);
        rs = pst.executeQuery();
        saved.setModel(DbUtils.resultSetToTableModel(rs));
        }
        catch (Exception e){
            JOptionPane.showMessageDialog(null, e);
        }
          finally
    {
      try
      {
          rs.close();
          pst.close();
      if(conn != null)
          conn.close();
      }
      catch(SQLException e)
      {

        System.err.println(e);
      }
    }
```

```
    }

    private void initComponents() {

    }

    private void browse3ActionPerformed(java.awt.event.ActionEvent evt) {
        new Browse().setVisible(true);
        this.setVisible(false);
    }

    private void logout3ActionPerformed(java.awt.event.ActionEvent evt) {
        new LogIn().setVisible(true);
        this.setVisible(false);
    }

    private void myopportunities3ActionPerformed(java.awt.event.ActionEvent evt) {
        new MyOpportunities().setVisible(true);
        this.setVisible(false);
    }

    private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
        new Contacts().setVisible(true);
        this.setVisible(false);
    }

    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Contacts().setVisible(true);
            }
        });
    }
}
```

```
import java.awt.HeadlessException;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.table.TableModel;
import net.proteanit.sql.DbUtils;

public class showCon extends javax.swing.JFrame {

    Connection conn = null;
    ResultSet rs = null;
    PreparedStatement pst = null;

    public showCon() {
        initComponents();
        try {
            conn = database.ConnecrDb();
        } catch (SQLException ex) {
            Logger.getLogger(Browse.class.getName()).log(Level.SEVERE, null, ex);
        }
        updateTable();
    }

    private void updateTable(){
        try{
        String sql = "SELECT FirstName, LastName, Email, Username, Status FROM Info";
        pst = conn.prepareStatement(sql);
        rs = pst.executeQuery();
        users.setModel(DbUtils.resultSetToTableModel(rs));
        }
        catch (Exception e){
            JOptionPane.showMessageDialog(null, e);
        }
          finally
    {
      try
      {
          rs.close();
          pst.close();
      if(conn != null)
          conn.close();
      }
      catch(SQLException e)
      {
```

```
        System.err.println(e);
      }
    }

    }
    private void initComponents() {

    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
this.setVisible(false);        // TODO add your handling code here:
    }

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        int index = users.getSelectedRow();
        TableModel model = users.getModel();

        String fname = model.getValueAt(index, 0).toString();
        String lname = model.getValueAt(index, 1).toString();
        String email = model.getValueAt(index, 2).toString();
        String username = model.getValueAt(index, 3).toString();
        String status = model.getValueAt(index, 4).toString();


            try {
            Class.forName("org.sqlite.JDBC");
        } catch (ClassNotFoundException ex) {
            Logger.getLogger(CreateOpportunity.class.getName()).log(Level.SEVERE,
null, ex);
        }
            Connection conn = null;
try{

            String url =
"jdbc:sqlite:C:\\Users\\Mario\\Documents\\NetBeansProjects\\IA\\mydatabase.sqlite";
             conn = DriverManager.getConnection(url);

            String sql = ("INSERT INTO Contacts (FirstName, LastName, Email,
Username, Status) "
                    + "VALUES ('"+fname+"', '"+lname+"', '"+email+"', '"+username+"',
'"+status+"')");

                  PreparedStatement preparedStmt = conn.prepareStatement(sql);
                    preparedStmt.execute();


        preparedStmt.close();
        }
        catch (HeadlessException | SQLException e){
            JOptionPane.showMessageDialog(null, e);
        }
```

```
finally
    {
      try
      {
        if(conn != null)
          conn.close();
      }
      catch(SQLException e)
      {

        System.err.println(e);
      }
    }
    }
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new showCon().setVisible(true);
            }
        });
    }
}
```