Contacts Management System in Netbeans and MySQL (Java Swing)

Raman Deep Singh

Download and Install MySQL Server from www.wampserver.com

Open MySQL Console and create database contacts.

Following are the sql queries to create database and table in contacts database;

create database contacts;

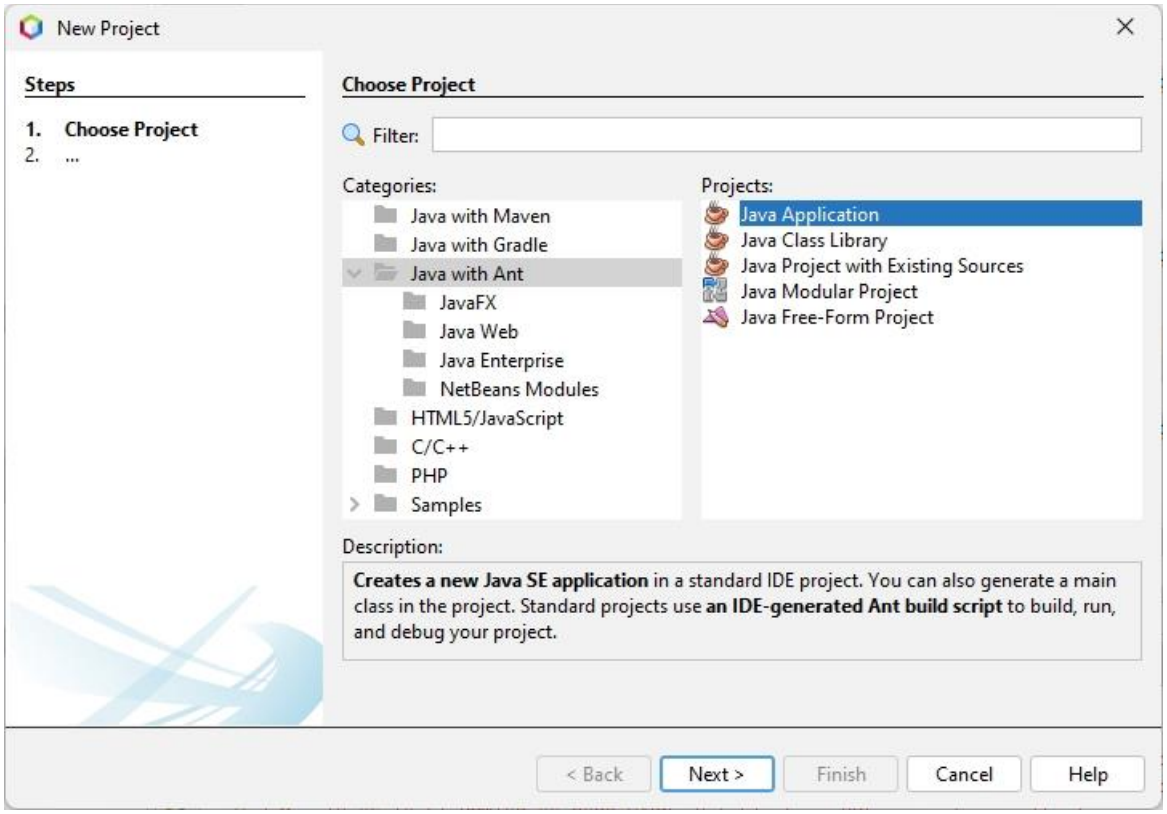
use contacts;

create table contacts(cname varchar(20),caddress varchar(20),cmobileno varchar(20),cemail varchar(20));

Start Apache Netbeans

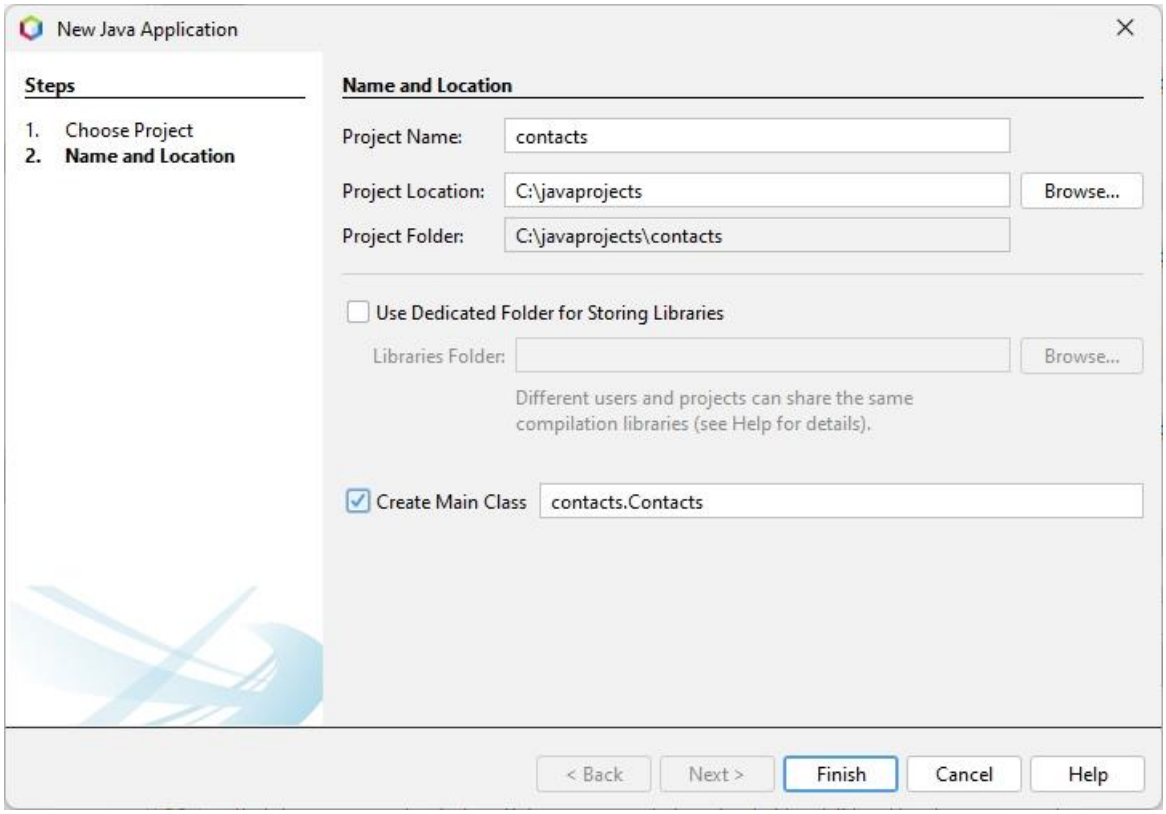
Click on File -> New Project

Following Window will appear



Click on Next

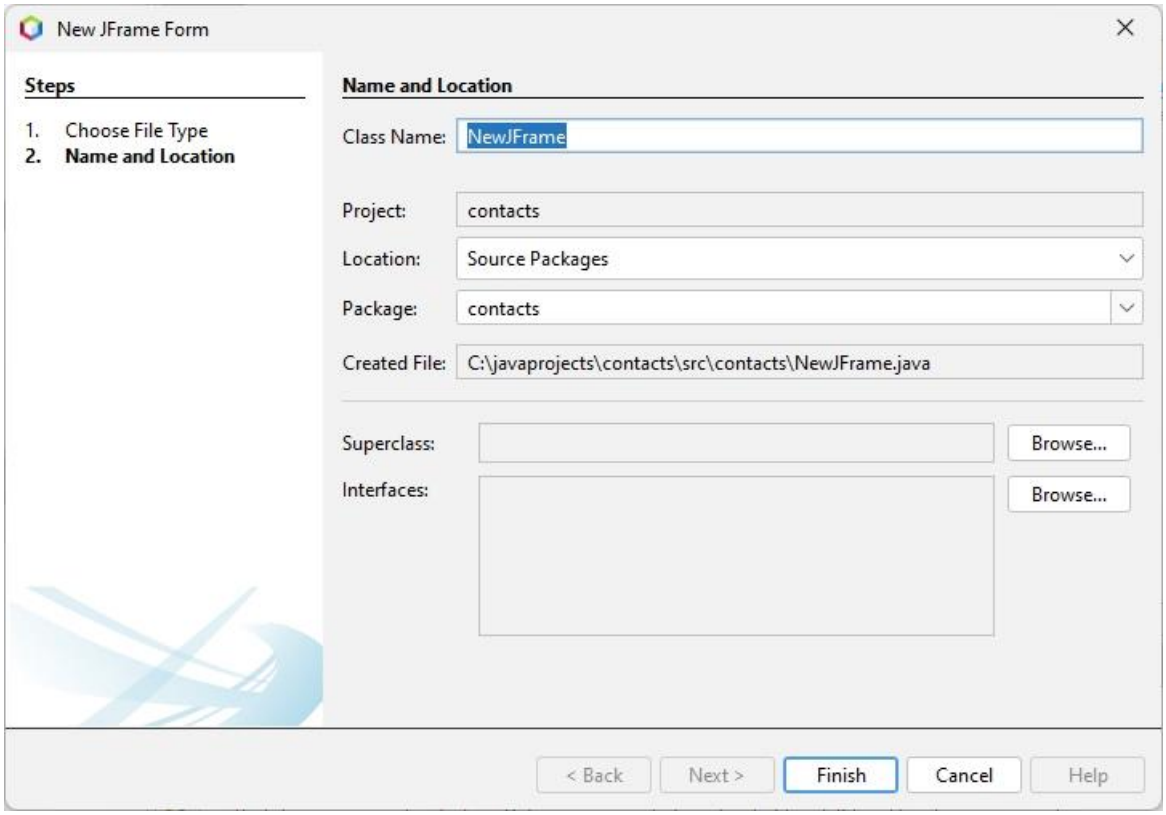
Following window will appear



Click Finish and after Clicking Finish a new project with name as contacts will be created in netbeans.

Goto contacts->Source Packages and under package contacts , create a new JFrame with name NewJFrame.

Following window will appear

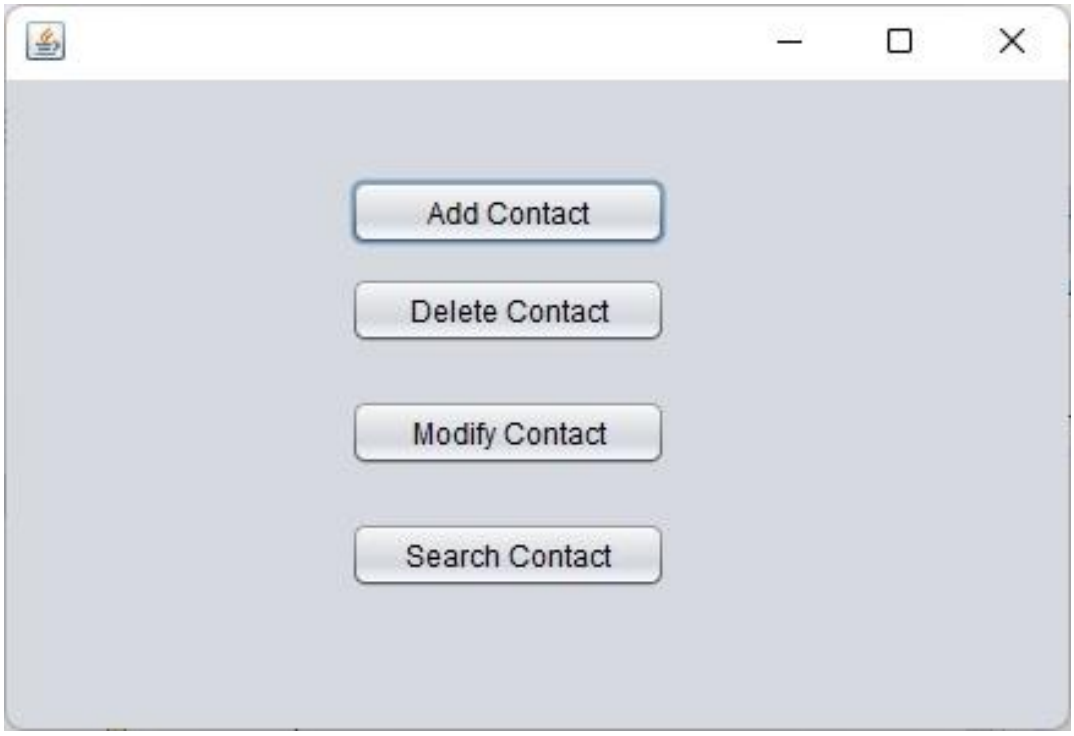


Click Finish

A New Frame will be created with name NewJFrame.

Goto Design View of the Frame and Right Click on the Design View and Select Layout as Absolute Layout. Absolute Layout means you can freely drag controls on the frame using design view.

Create Four Buttons on Frame



Following is the code for NewJFrame

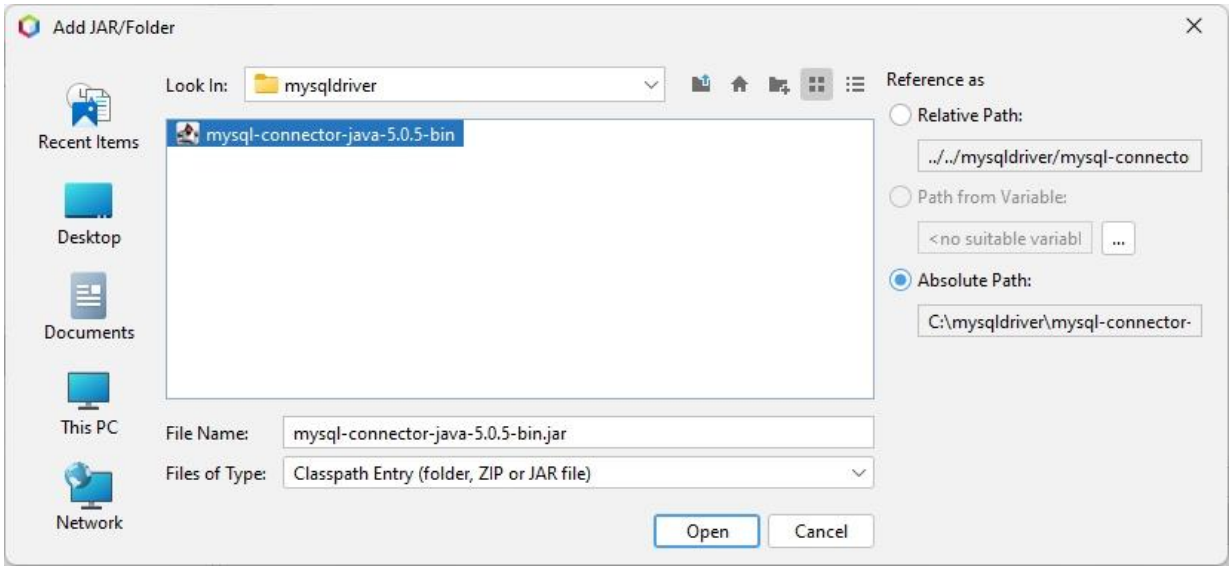
```
  private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
NewJFrame3 a=new NewJFrame3();
a.setSize(800,800);
a.setVisible(true);// TODO add your handling code here:
    }


    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
 NewJFrame2 a=new NewJFrame2();
a.setSize(800,800);
a.setVisible(true);       // TODO add your handling code here:
    }


    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
```

```
 NewJFrame1 a=new NewJFrame1();
a.setSize(800,800);
a.setVisible(true);        // TODO add your handling code here:
   }

   private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
 NewJFrame4 a=new NewJFrame4();
a.setSize(800,800);
a.setVisible(true);       // TODO add your handling code here:
   }
```

Now Add MySQL Connector Jar File to Libraries Folder

Right Click on Libraries Folder and Select option Add JAR/Folder

and Add MySQL Driver or Connector Jar File

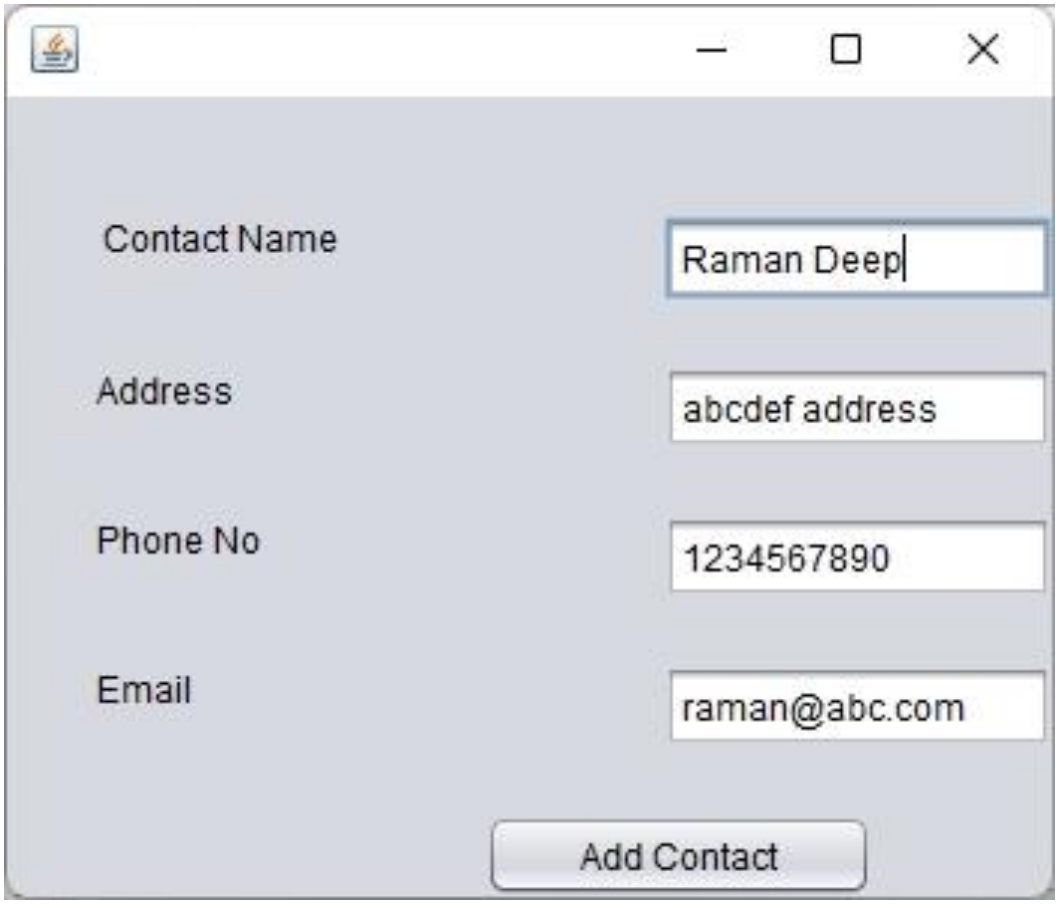


Click on Open and jar file will be added to your project.

MySQL Connector Jar file is a necessary file to connect Java Application to MySQL.

Now Create another JFrame with name NewJFrame1

Following is the design for the frame.



Following are the imports for the frame.

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
import javax.swing.JOptionPane;
```

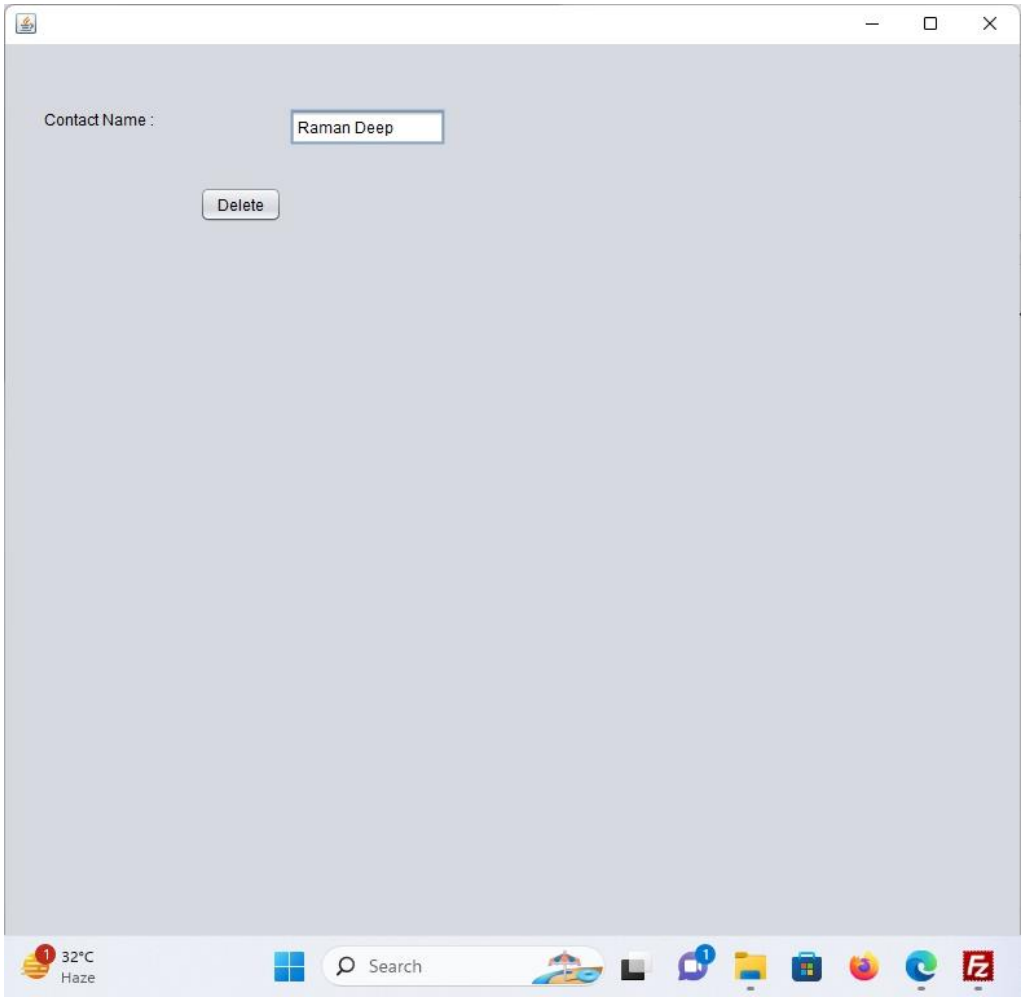
Following is the code for Button Add Contact.

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
try
   {

                        String driverName = "com.mysql.jdbc.Driver";
                        Class.forName(driverName);
String url ="jdbc:mysql://127.0.0.1:3306/contacts";
Statement stmt;

              String cname;
```

```
String caddress;
String cmobileno;
String cemail;

cname=jTextField1.getText();
caddress=jTextField2.getText();
cmobileno=jTextField3.getText();
cemail=jTextField4.getText();
        Connection con =DriverManager.getConnection(url,"root", "");
          // Create a connection to the database

         int rs;
          stmt = con.createStatement();
                rs = stmt.executeUpdate("insert into contacts values('"  + cname  + "','" +
caddress + "','" + cmobileno + "','" + cemail + "')");
                JOptionPane.showMessageDialog(null,"Contact Record Successfully Saved");
                }
                catch(Exception e)
                {
                System.out.println(e.toString());
                }       // TODO add your handling code here:
  }
```

Now Add another JFrame with name NewJFrame2

Following is the design for NewJFrame2.



Following are the imports for NewJFrame2

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
import javax.swing.JOptionPane;
```

Following is the code for Button Delete

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
 try{
      String driverName = "com.mysql.jdbc.Driver";
                          Class.forName(driverName);
String url ="jdbc:mysql://127.0.0.1:3306/contacts";
Statement stmt;
                    int rs;
              String cid;
```

```
cid=jTextField1.getText();
        Connection con =DriverManager.getConnection(url,"root", "");
                String cname;
      cname=jTextField1.getText();

          stmt = con.createStatement();
          int g;
          g=stmt.executeUpdate("delete from contacts where cname='"+cname + "'");
JOptionPane.showMessageDialog(null,"Record Successfully Deleted");

                }
                catch(Exception e)
                {
                System.out.println(e.toString());
                }       // TODO add your handling code here:
  }
```

Now Add another frame with name as NewJFrame3

Design for NewJFrame3 is

Imports for NewJFrame3 are

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.Statement;
import javax.swing.JOptionPane;
```

Following is the code for Modify Record Button is

```
  private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
 try{

                                    String driverName = "com.mysql.jdbc.Driver";
                                     Class.forName(driverName);
String url ="jdbc:mysql://127.0.0.1:3306/contacts";
Statement stmt;
                                  int rs;
                                  Connection con =DriverManager.getConnection(url,"root", "");
String cname=jTextField1.getText();
```

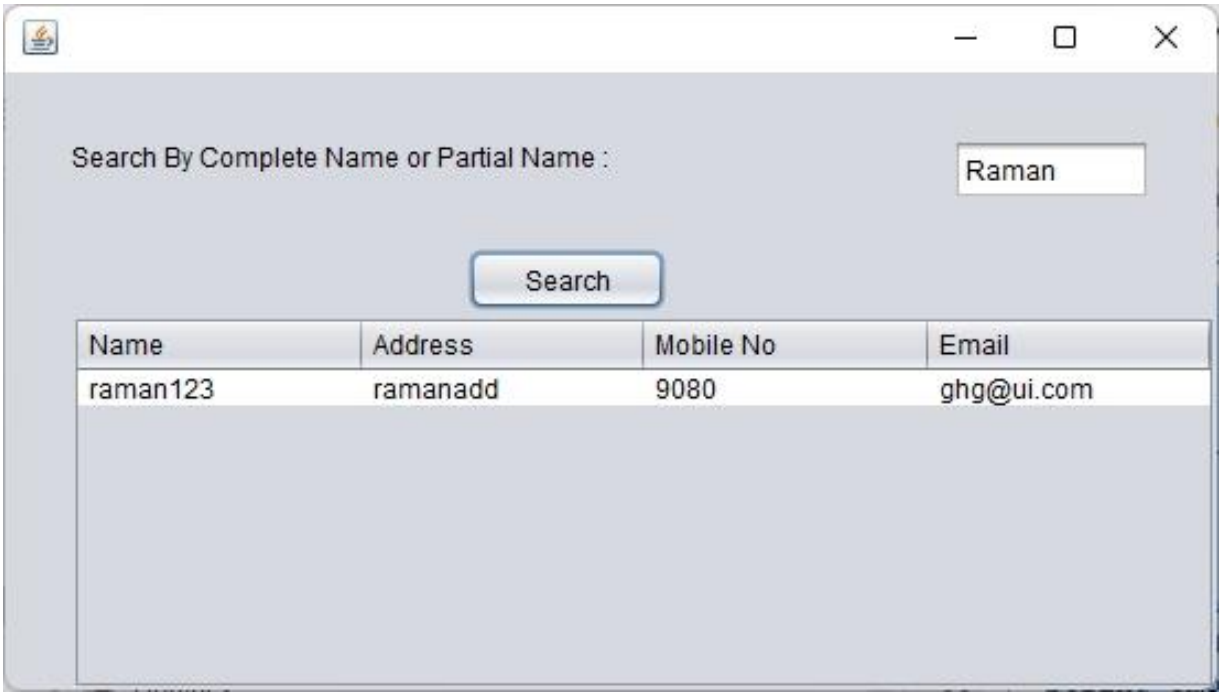
```
String caddress=jTextField2.getText();
String cmobileno=jTextField3.getText();
String cemail=jTextField4.getText();



                                stmt = con.createStatement();
                                int g;
                                g=stmt.executeUpdate("update contacts set caddress='"+caddress+
"',cmobileno='" + cmobileno +  "',cemail='"+ cemail + "' where cname='"+cname + "';");
                            if(g>0)
      {
         JOptionPane.showMessageDialog(null,"Record Successfully Updated");
      }
      else
      {
         JOptionPane.showMessageDialog(null,"Record Can't Be Updated");
      }


                                        }
                                        catch(Exception e)
                                        {
                                        System.out.println(e.toString());
                                        }       // TODO add your handling code here:
   }
```

Now add another frame with name NewJFrame4

you will have to place a JTable control for search frame

before adding JTable control place a JScrollPane control from Swing Containers Panel and on this panel place a JTable so that JTable has horizontal and vertical scrollbar.

Design for NewJframe4 is as given



Following are the imports for NewJFrame4

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.table.DefaultTableModel;
```

Following is the code for Search Button

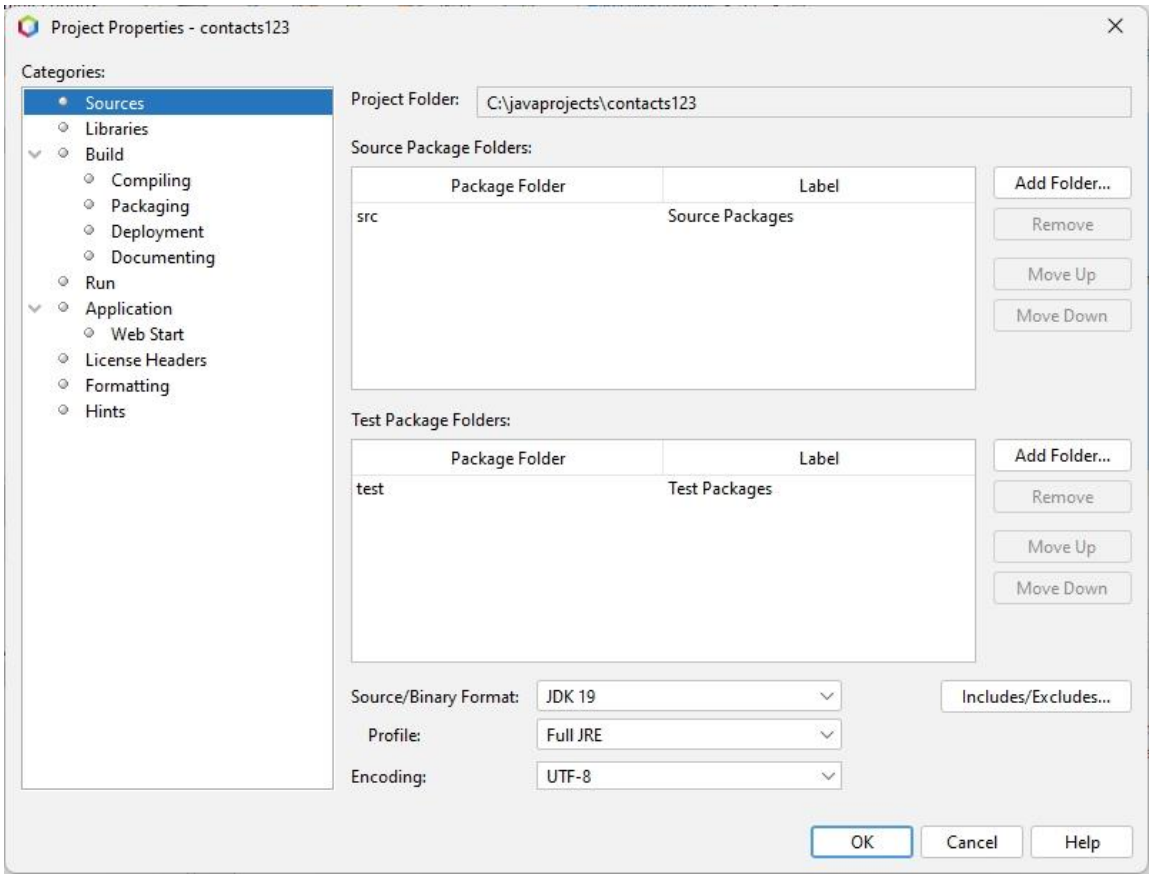
```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
try
 {
    DefaultTableModel model = new DefaultTableModel();
 String[] columnNames = {"Name" ,"Address","Mobile No","Email"};
model.setColumnIdentifiers(columnNames);
jTable1.setModel(model);
String cname=jTextField1.getText();

Statement stmt;
                        ResultSet rs;
```

```
                Class.forName("com.mysql.jdbc.Driver");
                String url
="jdbc:mysql://127.0.0.1:3306/contacts?useUnicode=true&useJDBCCompliantTimezoneShift=true&
useLegacyDatetimeCode=false&serverTimezone=UTC";
                Connection con =DriverManager.getConnection(url,"root", "");
                stmt = con.createStatement();
                rs = stmt.executeQuery("SELECT * from contacts where cname like '%" + cname +
"%'");




        String contactname;
        String caddress;
        String cmobileno;
        String cemail;


                while(rs.next()){
        contactname=rs.getString("cname");
        caddress=rs.getString("caddress");
        cmobileno=rs.getString("cmobileno");
        cemail=rs.getString("cemail");
        model.addRow(new Object[]{contactname,caddress,cmobileno,cemail});
                        }
}
                catch(Exception e)
                {
                System.out.println(e.toString());
                }       // TODO add your handling code here:
   }
```

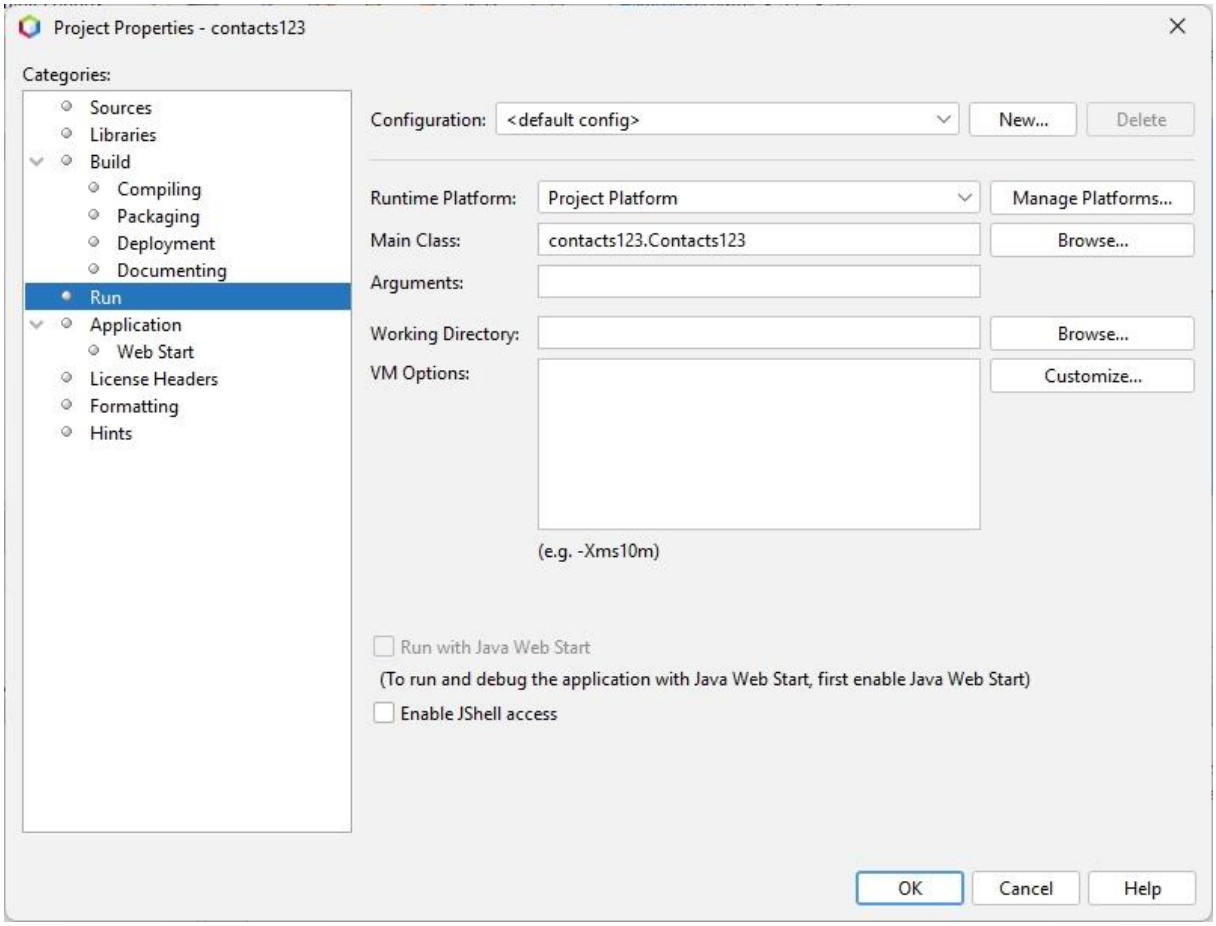
Now to ensure that when you run your program first frame displayed should be NewJFrame following selection needs to be done

Right Click on project contacts and select option Properties.

Following window will appear

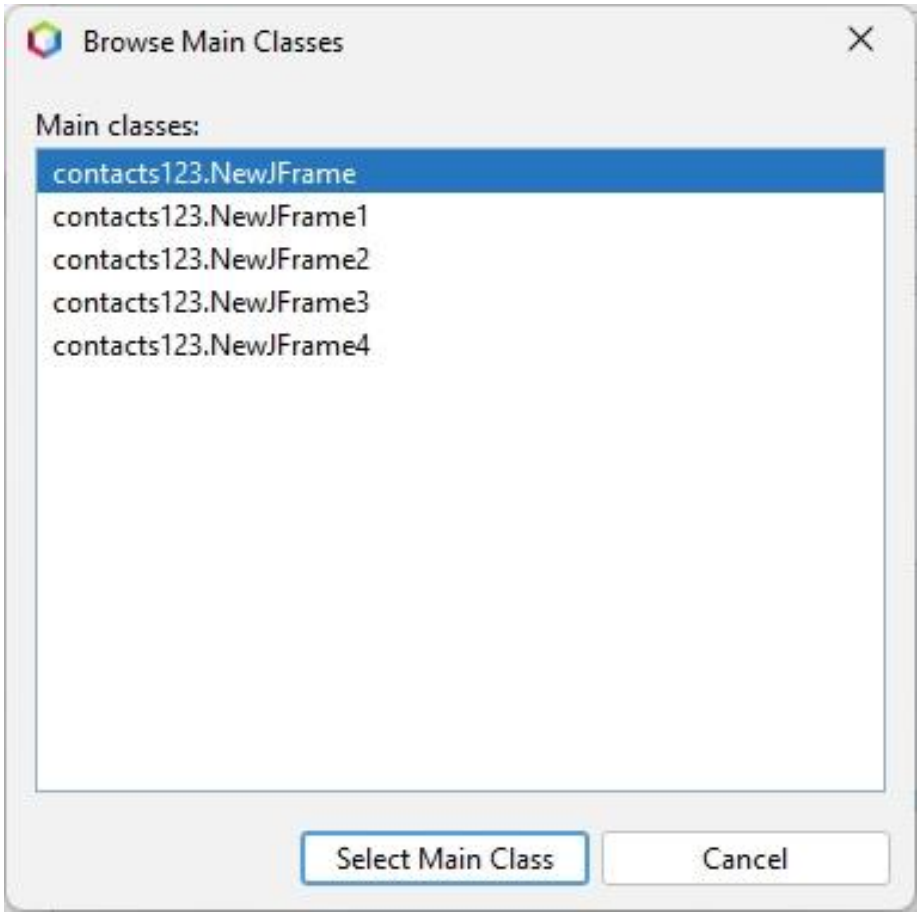


Go to Run and Following window will appear.



Click on Browse Button next to Main Class and select the form NewJFrame.

Following window shows the method to select NewJFrame as Main Class which means the starting class.



Select Main Class and default running class will be set to NewJFrame.

Project is complete right click on project and select Build , this will compile the project and run the project after clicking Run.

After Successful Run following window will appear.