SOURCE CODE

Class 1

```
package jdbcdemo;


import java.awt.*;

import java.awt.event.*;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.awt.FlowLayout;
import java.sql.*;
import javax.swing.*;

import
net.sf.dynamicreports.jasper.builder.JasperReportBuilder;
import net.sf.dynamicreports.report.builder.DynamicReports;
import net.sf.dynamicreports.report.builder.column.Columns;
import
net.sf.dynamicreports.report.builder.component.Components;
import
net.sf.dynamicreports.report.builder.datatype.DataTypes;
import
net.sf.dynamicreports.report.constant.HorizontalTextAlignment;
import net.sf.dynamicreports.report.exception.DRException;

public class Driver extends JFrame implements ActionListener
{
     ass z = new ass();
//   test y = new test();
     static JLabel label1,label2,label3,sep;
     static JButton
sign,create,deposit,withdraw,trans,view,close,clear,delete,rep
ort;
     static JTextField txtA1;
     static JTextField txtA2;
     static Connection con;
     static Statement st1,st2,st3;
     static ResultSet rs1;
     int numd;
     int numf;
     int nume;
```

```
     public Driver()
     {
          super("Bank");
          setLayout(new FlowLayout());
          try {

               con = DriverManager.getConnection
("jdbc:mysql://localhost:3306/bank", "root", "Iamthehero123");
               st1=con.createStatement();
               rs1=st1.executeQuery("select * from
account_table");
          }
          catch(Exception e)
          {
               System.out.println("there was some error in
establishing connection : "+e);
          }
          sep = new JLabel("Existing User Signin  :  ");
          add(sep);
          label1 = new JLabel(" Account Number : ");
          add(label1);
          txtA1 = new JTextField(15);
          add(txtA1);
          label2 = new JLabel("        Name :       ");
          add(label2);
          txtA2 = new JTextField(15);
          add(txtA2);
          sign=new JButton("SIGN IN");
          add(sign);
          sep = new JLabel("
");
          add(sep);
          sep = new JLabel("
");
          add(sep);
          sep = new JLabel("New User Signin  :  ");
          add(sep);
          create=new JButton("CREATE ACCOUNT");
          add(create);
          sep = new JLabel("
");
          add(sep);
          sep = new JLabel("
");
          add(sep);
          deposit=new JButton("DEPOSIT");
          add(deposit);
          withdraw=new JButton("WITHDRAW");
          add(withdraw);
```

```
            trans=new JButton("TRANSACTION");
            add(trans);
            view=new JButton("VIEW ACCOUNT");
            add(view);
            clear=new JButton("CLEAR");
            add(clear);
            delete=new JButton("DELETE ACCOUNT");
            add(delete);
            report=new JButton("GENERATE REPORT");
            add(report);
            sep = new JLabel("
");
            add(sep);
            close=new JButton("CLOSE");
            add(close);
            create.addActionListener(this);
            deposit.addActionListener(this);
            withdraw.addActionListener(this);
            trans.addActionListener(this);
            view.addActionListener(this);
            clear.addActionListener(this);
            delete.addActionListener(this);
            report.addActionListener(this);
            close.addActionListener(this);
            sign.addActionListener(this);
            deposit.setEnabled(false);
            withdraw.setEnabled(false);
            trans.setEnabled(false);
            view.setEnabled(false);
            delete.setEnabled(false);
            report.setEnabled(false);
      }

      public static boolean isNumeric(String str)
      {
        return str.matches("-?\\d+(\\.\\d+)?");  //match a
number with optional '-' and decimal.
      }

      public void actionPerformed(ActionEvent e)
      {
            if(e.getSource()==create)
            {
                  String stra =
JOptionPane.showInputDialog(null,"Please enter the NAME to
open a CURRENT ACCOUNT");
                  String strb =
JOptionPane.showInputDialog(null,"Please enter the ACCOUNT
NUMBER");
```

```
                if((stra==null || strb==null) ||
(isNumeric(strb)==false))
                {
                     JOptionPane.showMessageDialog(null,"
Either any one or both the values were left blank or account
no. not valid");
                     deposit.setEnabled(false);
                     withdraw.setEnabled(false);
                     trans.setEnabled(false);
                     view.setEnabled(false);
                     create.setEnabled(true);
                     sign.setEnabled(true);
                     delete.setEnabled(false);
                     report.setEnabled(false);
                }
                else
                {
                     try
                     {
                          st2=con.createStatement();

                          JOptionPane.showMessageDialog(null,
stra);

                          z.banks(1, Integer.parseInt(strb),
stra, 0, 0, 0);

                          st2.close();


                          txtA1.setText(strb);
                          txtA2.setText(stra);
                     }
                     catch(Exception e2)
                     {

                     }
                     deposit.setEnabled(true);
                     withdraw.setEnabled(true);
                     trans.setEnabled(true);
                     view.setEnabled(true);
                     create.setEnabled(true);
                     sign.setEnabled(true);
                     delete.setEnabled(true);
                }
           }
           else if(e.getSource()==sign)
           {
                String num=txtA1.getText();
```

```
                try
                {
                     st1.close();
                     st1=con.createStatement();
                     rs1=st1.executeQuery("Select * from
account_table where AccNum Like '"+num+"'");
                     JOptionPane.showMessageDialog(null,"Sign
in successful");
                     rs1.next();
                     txtA1.setText(rs1.getString("AccNum"));
                     txtA2.setText(rs1.getString("UserName"));
                }
                catch(Exception e2)
                {
                     if(num.equals("0000") &&
txtA2.getText().equals("Akash")){

     JOptionPane.showMessageDialog(null,"Admin Access
Granted");
                     }
                     else{

     JOptionPane.showMessageDialog(null,"Invalid Account
Number");
                     }
                }
                deposit.setEnabled(true);
                withdraw.setEnabled(true);
                trans.setEnabled(true);
                view.setEnabled(true);
                create.setEnabled(false);
                delete.setEnabled(true);
                if(num.equals("0000") &&
txtA2.getText().equals("Akash")){
                     report.setEnabled(true);
                }
           }
           else if(e.getSource()==deposit)
           {
                String strc =
JOptionPane.showInputDialog(null,"Please enter the amount to
be DEPOSITED");
                numd=Integer.parseInt(strc);
                int numb=Integer.parseInt(txtA1.getText());
                if (numd<=0)
                {
                     JOptionPane.showMessageDialog(null," NO
Amount Deposited");
                }
```

```
                else
                {
                     try
                     {
                          int num,num1;
                          st3=con.createStatement();
                          rs1=st3.executeQuery("select * from
account_table where AccNum like '"+txtA1.getText()+"'");
                          rs1.next();


     num1=Integer.parseInt(rs1.getString("Deposit"));

     num=Integer.parseInt(rs1.getString("Balance"));

     JOptionPane.showMessageDialog(null,"Amount Deposited
Successfully , Amount is "+numd);

                          numf=num+numd;
                          num1=num1+numd;

                          z.banks(2, numb, txtA2.getText(),
num1, numf, 0);


                          st3.close();
                     }
                     catch(Exception e2)
                     {

     JOptionPane.showMessageDialog(null,"Amount can not be
DEPOSITED");
                     }
                }
           }
           else if(e.getSource()==withdraw)
           {
                String strd =
JOptionPane.showInputDialog(null,"Please enter the amount to
be WITHDRAWN");
                nume=Integer.parseInt(strd);
                int numc=Integer.parseInt(txtA1.getText());


                     try
                     {
                          int num1;
                          st3=con.createStatement();
```

```
                         rs1=st3.executeQuery("select * from
account_table where Accnum like '"+txtA1.getText()+"'");
                         rs1.next();
                         numf =
Integer.parseInt(rs1.getString("Balance"));
                         if(numf<=nume)
                         {

     JOptionPane.showMessageDialog(null,"Sorry can not
Withdraw, no sufficient Balance");
                         }
                         else
                         {


     num1=Integer.parseInt(rs1.getString("Withdraw"));

     JOptionPane.showMessageDialog(null,"Amount Withdrawn is
"+nume);
                         numf=numf-nume;
                         num1=num1+nume;

                         z.banks(3, numc, txtA2.getText(), 0,
numf, num1);

                    }}
                    catch(Exception e2)
                    {
                         JOptionPane.showMessageDialog(null,"
Can't WITHDRAW ");
                    }
               }

          else if(e.getSource()==view)
          {
               String num=txtA1.getText();

               try
               {

                    st3=con.createStatement();
                    rs1=st3.executeQuery("select * from
account_table where Accnum like '"+num+"'");
                    rs1.next();

     numf=Integer.parseInt(rs1.getString("AccNum"));
                    String userName =
rs1.getString("UserName");
```

```
                    int dep =
Integer.parseInt(rs1.getString("Deposit"));
                    int bal =
Integer.parseInt(rs1.getString("Balance"));
                    int with =
Integer.parseInt(rs1.getString("Withdraw"));


     JOptionPane.showMessageDialog(null,"Account is: "+numf +
                                "\nUsername: "+userName +
"\nDeposit: "+dep+"\nBalance: "+bal+
                                "\nWithdraw: "+with);
               }
               catch(Exception e2)
               {
                    JOptionPane.showMessageDialog(null,"
Balance null ");
               }

          }
          else if(e.getSource()==trans)
          {
               String strq =
JOptionPane.showInputDialog(null,"Enter the ACCOUNT NUMBER to
view the TRANSACTION DONE");
               int numk=Integer.parseInt(strq);
               int numa=Integer.parseInt(txtA1.getText());
               if(numk==numa)
               {
                    JOptionPane.showMessageDialog(null,"Amount
Deposited  is "+numd);
                    JOptionPane.showMessageDialog(null,"Amount
Withdrawn is "+nume);
               }
               else
               {
                    JOptionPane.showMessageDialog(null,"The
Account Number is not the same as in the TEXTFIELD");
               }
          }

          else if(e.getSource()==delete)
          {
               String strq =
JOptionPane.showInputDialog(null,"Enter the ACCOUNT NUMBER to
DELETE");
               int numk=Integer.parseInt(strq);
               int numa=Integer.parseInt(txtA1.getText());
               if(numk==numa)
```

```
                {
                     z.banks(4, numa, "", 0, 0, 0);
                }
                else
                {
                     JOptionPane.showMessageDialog(null,"The
Account Number is not the same as in the TEXTFIELD");
                }
           }

           else if(e.getSource()==report)
           {
                Connection connection = null;
                try {
                     Class.forName("com.mysql.jdbc.Driver");
                     connection = DriverManager.getConnection(

     "jdbc:mysql://localhost:3306/bank", "root",
"Iamthehero123");
                } catch (SQLException k) {
                     k.printStackTrace();
                     return;
                } catch (ClassNotFoundException k) {
                     k.printStackTrace();
                     return;
                }

                JasperReportBuilder report =
DynamicReports.report();//a new report
                report
                  .columns(
                      Columns.column("Accoun No.", "AccNum",
DataTypes.integerType()),
                      Columns.column("Username", "UserName",
DataTypes.stringType()),
                      Columns.column("Deposit", "Deposit",
DataTypes.integerType()),
                      Columns.column("Balance", "Balance",
DataTypes.integerType()),
                      Columns.column("Withdraw", "Withdraw",
DataTypes.integerType()))
                  .title(//title of the report
                      Components.text("Bank Management Report")

.setHorizontalTextAlignment(HorizontalTextAlignment.CENTER))
                       .pageFooter(Components.pageXofY())//show
page number on the page footer
                       .setDataSource("SELECT AccNum, UserName,
Deposit, Balance, Withdraw FROM account_table",
```

```
                                                  connection);

                try {
                              //show the report
                     report.show();

                              //export the report to a pdf file
                     report.toPdf(new
FileOutputStream("c:/report.pdf"));
                } catch (DRException k) {
                     k.printStackTrace();
                } catch (FileNotFoundException k) {
                     k.printStackTrace();
                }



          }

          else if(e.getSource()==clear)
          {
                txtA1.setText("");
                txtA2.setText("");
                deposit.setEnabled(false);
                withdraw.setEnabled(false);
                trans.setEnabled(false);
                view.setEnabled(false);
                create.setEnabled(true);
                sign.setEnabled(true);
                delete.setEnabled(false);
          }
          else if(e.getSource()==close)
          {
                System.exit(0);
          }
     }
     public static void main(String args[])
     {
          Driver obj = new Driver();
          obj.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
          obj.setSize(250,320);
          obj.setVisible(true);
     }}
```

Class 2

```
package jdbcdemo;

import java.sql.*;

import javax.swing.JOptionPane;
public class ass {

     public static void main(String[] args) {

     }

          public static void banks(int k, int i, String z, int
num1, int num2, int num3){
          try{
               //1. Get a connection to database
               Connection myConn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/bank"
, "root", "Iamthehero123");

               //2. Create a statement
               Statement myStmt = myConn.createStatement();

               if (k==1){
               String rr = "insert into account_table "
                         + " (AccNum, UserName, Deposit,
Balance, Withdraw)"
                         + " values ("+i+",'"
+z.toString()+"', 0, 0, 0)";


               myStmt.executeUpdate(rr);
               JOptionPane.showMessageDialog(null,"ACCOUNT
Created Successfully");

               }

               if (k==2) {
                    String ss="update account_table set
Deposit="+num1+" where AccNum="+i+"";
                    String aa="update account_table set
Balance="+num2+" where AccNum="+i+"";

                    myStmt.executeUpdate(ss);
                    myStmt.executeUpdate(aa);
               }
```

```
               if (k==3) {
                    String pp="update account_table set
Withdraw="+num3+" where AccNum="+i+"";
                    String pp1="update account_table set
Balance="+num2+" where AccNum="+i+"";

                    myStmt.executeUpdate(pp);
                    myStmt.executeUpdate(pp1);
               }

               if (k==4){
                    String rr = "delete from account_table
where AccNum="+i+"";


                    myStmt.executeUpdate(rr);

     JOptionPane.showMessageDialog(null,"ACCOUNT Deletion
Successfully");

                    }

//             //4. Process the result
               ResultSet myRs = myStmt.executeQuery("select *
from account_table");

               while(myRs.next()) {

     System.out.println(myRs.getString("AccNum") + ", " +
myRs.getString("UserName")+ ", " + myRs.getString("Deposit")+
", " + myRs.getString("Balance")+ ", " +
myRs.getString("Withdraw"));
          }
               }
          catch (Exception exc) {
               if (k==1) {
               JOptionPane.showMessageDialog(null," There is
already a user with same ACCOUNT NUMBER ,Give another ACCOUNT
NUMBER ");
               }
     }
     }
}
```