SRM UNIVERSITY

JAVA PROGRAMMING.

CSE 207 L

Project Synopsis on

"BANK MANAGEMENT SYSTEM."

Submitted in partial fulfilment for the award of the degree in

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY:

1: HARSHA VARDHINI.

AP21110011525

2: JEEVANANDHA REDDY

AP21110011527

3: RAMSAI ACHANTA

AP21110011554

4: MANOHAR PYLA

AP21110011520

Group-11, Section-V

Under the guidance of

Mr. NARALA SUDHAKAR REDDY.

CONTENTS

ABSTRACT

The Bank Management System project aims to simulate real-world banking operations. It provides hands-on experience of working with bank accounts, transactions, and user interactions, which are fundamental aspects of the banking industry. By developing this project, learners can gain practical exposure to the concepts and workflows employed in the banking domain. The aim of the project is to develop a simple Bank Management System that allows users to register, login, perform banking transactions such as deposit and withdrawal, check account balance, and delete their account if needed.

CHAPTER: 1

INTRODUCTION

The Bank Management System project can serve as a practical exercise for learning Java programming concepts such as classes, objects, inheritance, file handling, and user input/output. The project provides a foundation for building more advanced banking systems or integrating additional features such as transferring funds between accounts, generating account statements, or implementing security measures. The Bank Management System can be used as a teaching tool in computer science or programming courses to demonstrate software development techniques, file handling, and user interaction. Java is a widely used programming language in the banking and financial sector. By building the Bank Management System project, developers can enhance their Java programming skills, understand object-oriented programming principles, and gain familiarity with file handling operations. These skills are highly valuable in the software development industry. The project serves as an excellent educational tool for students studying computer science, software engineering, or related disciplines. It provides a concrete example of applying core programming concepts, such as classes, objects, and control flow, in a practical scenario. It also reinforces the understanding of input/output operations and persistence through file handling.

CHAPTER: 2

OBJECTIVE

- The Bank Management System is a Window-based application written in Java.

- It utilizes object-oriented programming principles & DataBaseManagmentSystem with SQL to provide basic banking functionalities.

- The system allows users to create bank accounts with automatic generation of unique account number, PIN, MICR number(where later PIN can be changed using change PIN option) &based on user’s personals like account holder name, initial balance, Date of birth, Address, Nationality, Mobile number, Security Question &an answer to it.

- Users can perform transactions on their accounts, including transfering depositing and withdrawing funds.

- They can also check their account balance and can change their account pin.

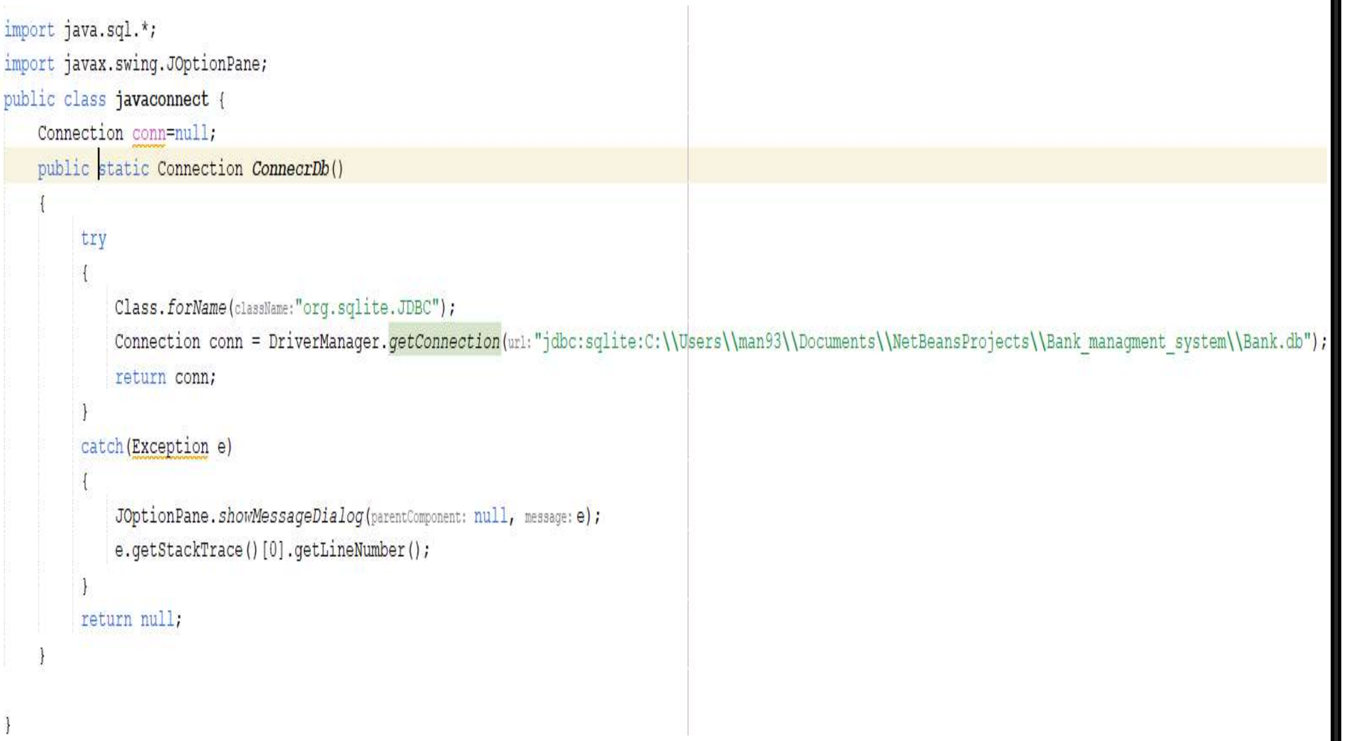
- The system stores account information in a Database which will be thoroughly updated using basic SQL commands.

CHAPTER: 3

SYSTEM IMPLEMENTATION

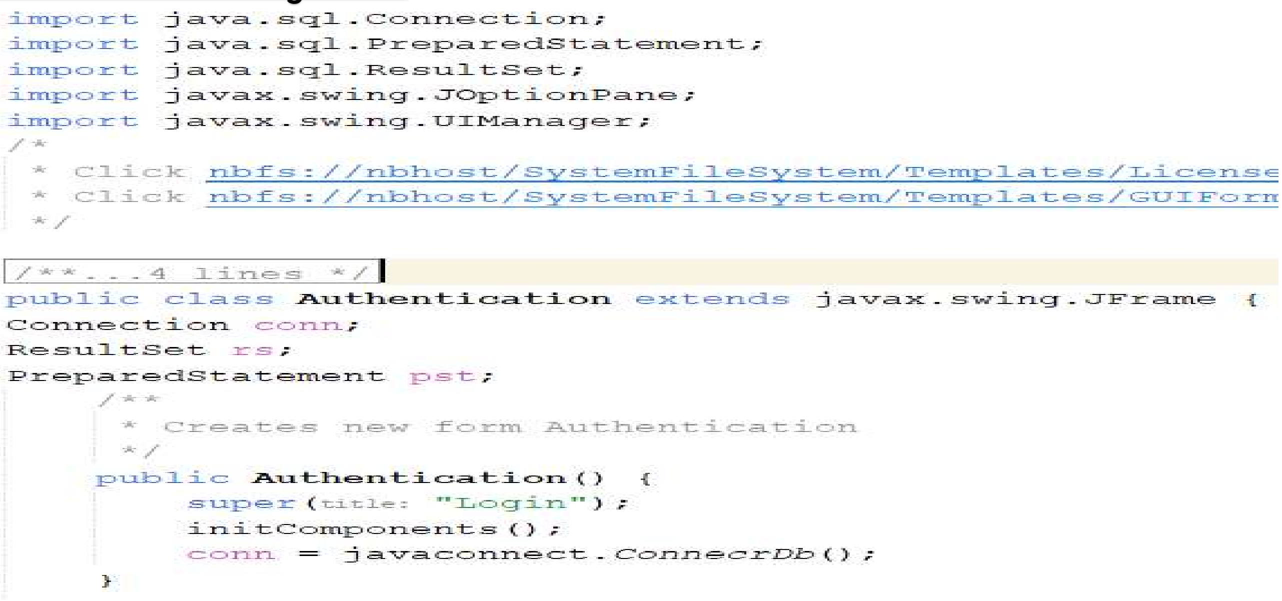
PROGRAM:

#connect.java code(to connect our java program with required Database file):



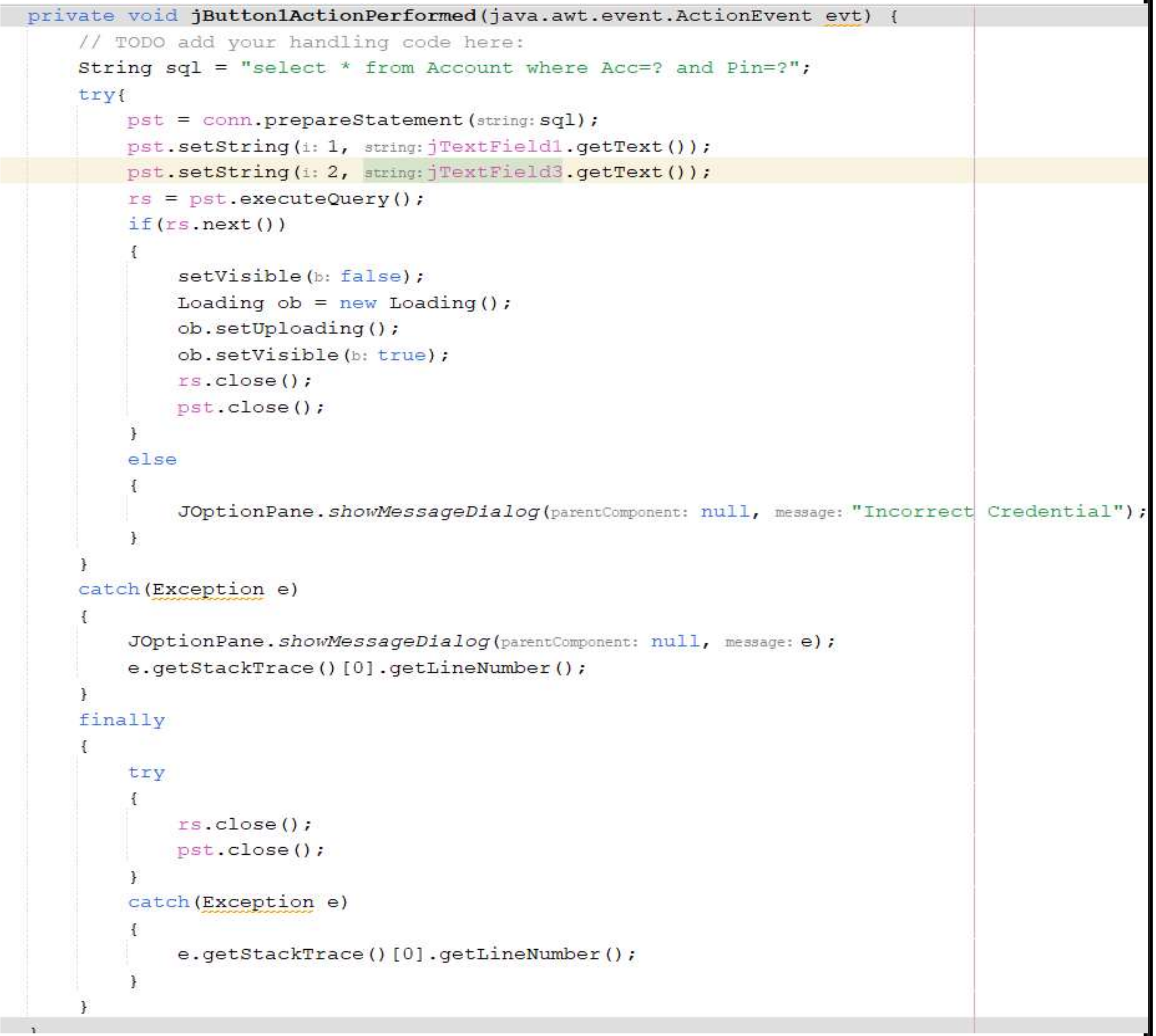
```
import java.sql.*;
import javax.swing.JOptionPane;
public class javaconnect {
    Connection conn=null;
    public static Connection ConnecrDb()
    {
        try
        {
            Class.forName(className:"org.sqlite.JDBC");
            Connection conn = DriverManager.getConnection(url:"jdbc:sqlite:C:\\Users\\man93\\Documents\\NetBeansProjects\\Bank_managment_system\\Bank.db");
            return conn;
        }
        catch(Exception e)
        {
            JOptionPane.showMessageDialog(parentComponent: null, message: e);
            e.getStackTrace()[0].getLineNumber();
        }
        return null;
    }


}
```

#Authentication Page Code:



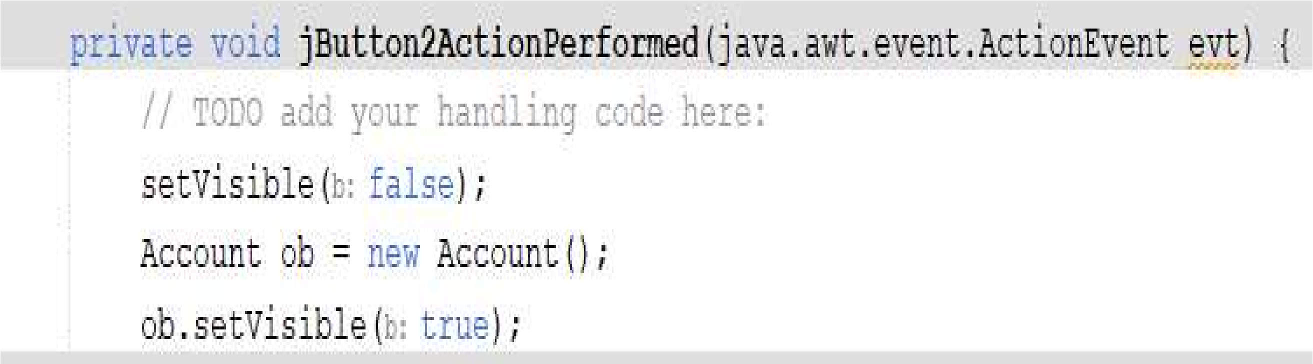
```
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.swing.JOptionPane;
import javax.swing.UIManager;
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/License
 * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForm
 */

/**...4 lines */
public class Authentication extends javax.swing.JFrame {
Connection conn;
ResultSet rs;
PreparedStatement pst;
    /**
     * Creates new form Authentication
     */
    public Authentication() {
        super(title: "Login");
        initComponents();
        conn = javaconnect.ConnecrDb();
    }
```

#Code for Login Button in Authentication Page:



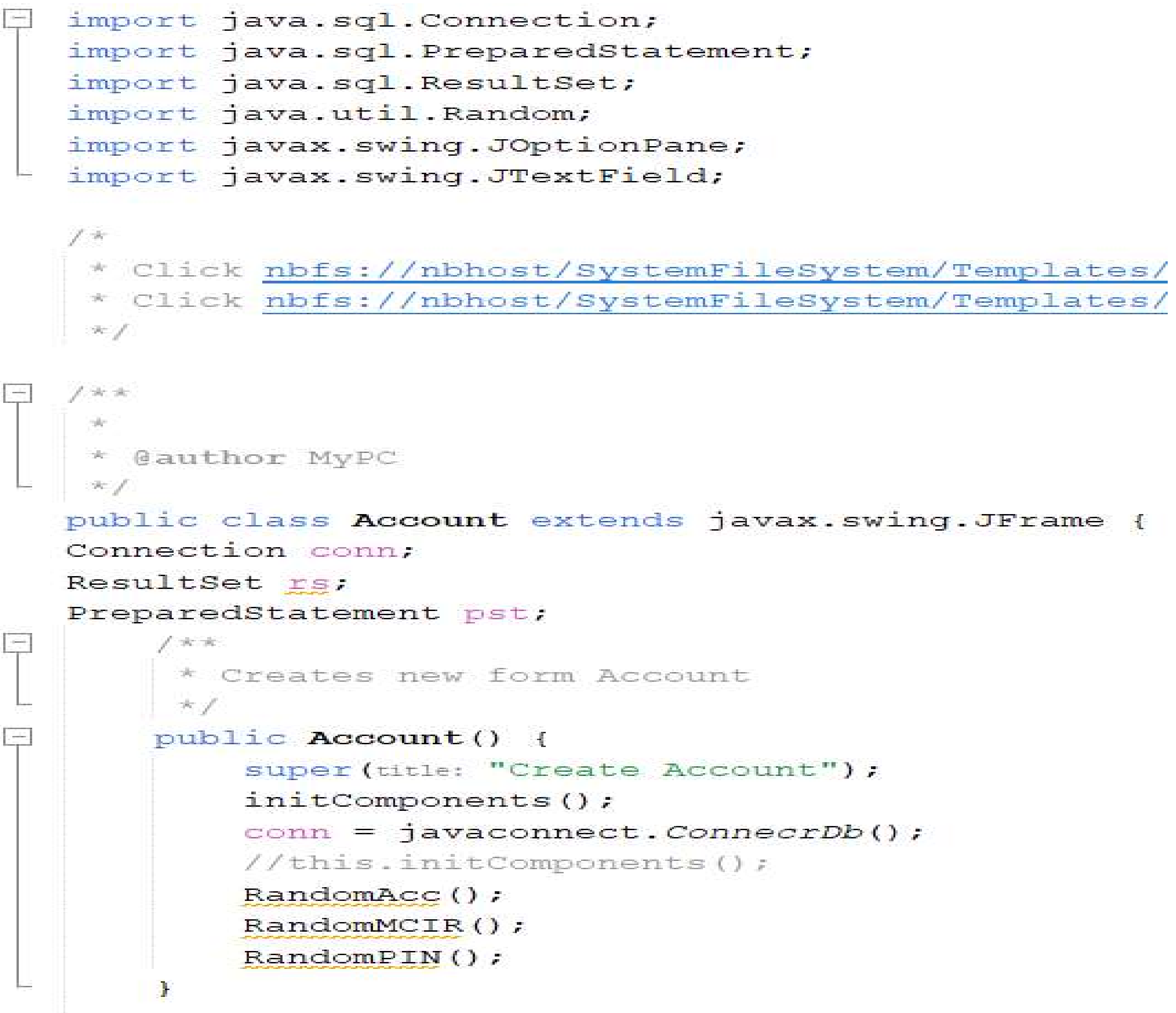
```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String sql = "select * from Account where Acc=? and Pin=?";
    try{
        pst = conn.prepareStatement(string: sql);
        pst.setString(i: 1, string: jTextField1.getText());
        pst.setString(i: 2, string: jTextField3.getText());
        rs = pst.executeQuery();
        if(rs.next())
        {
            setVisible(b: false);
            Loading ob = new Loading();
            ob.setUploading();
            ob.setVisible(b: true);
            rs.close();
            pst.close();
        }
        else
        {
            JOptionPane.showMessageDialog(parentComponent: null, message: "Incorrect Credential");
        }
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
        e.getStackTrace()[0].getLineNumber();
    }
    finally
    {
        try
        {
            rs.close();
            pst.close();
        }
        catch(Exception e)
        {
            e.getStackTrace()[0].getLineNumber();
        }
    }
}
```

#Code for Create Account Button in Authentication Page:



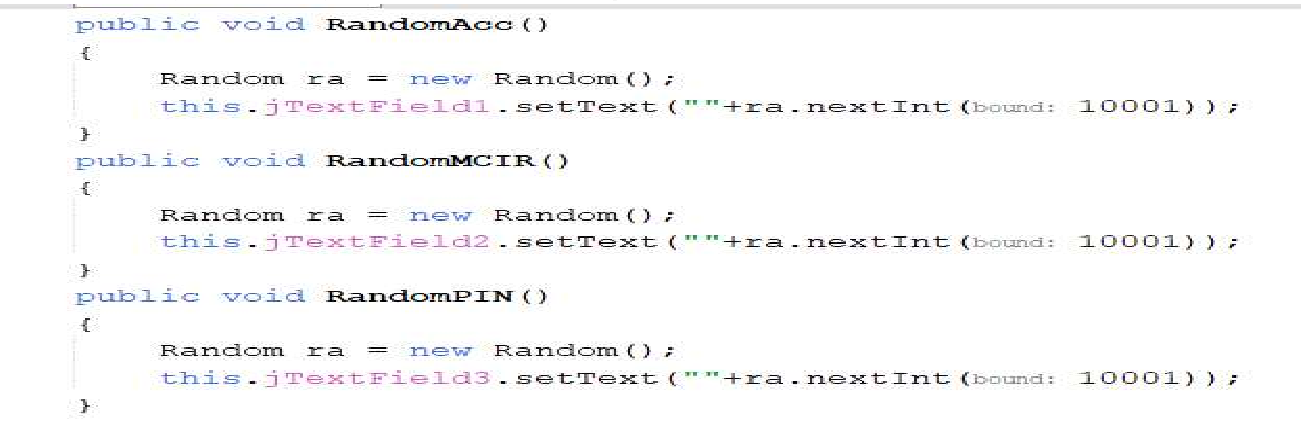
```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    setVisible(b: false);
    Account ob = new Account();
    ob.setVisible(b: true);
```

#Code for Account Page(for creating new Account, invoked by create Account button in Authentication Page):



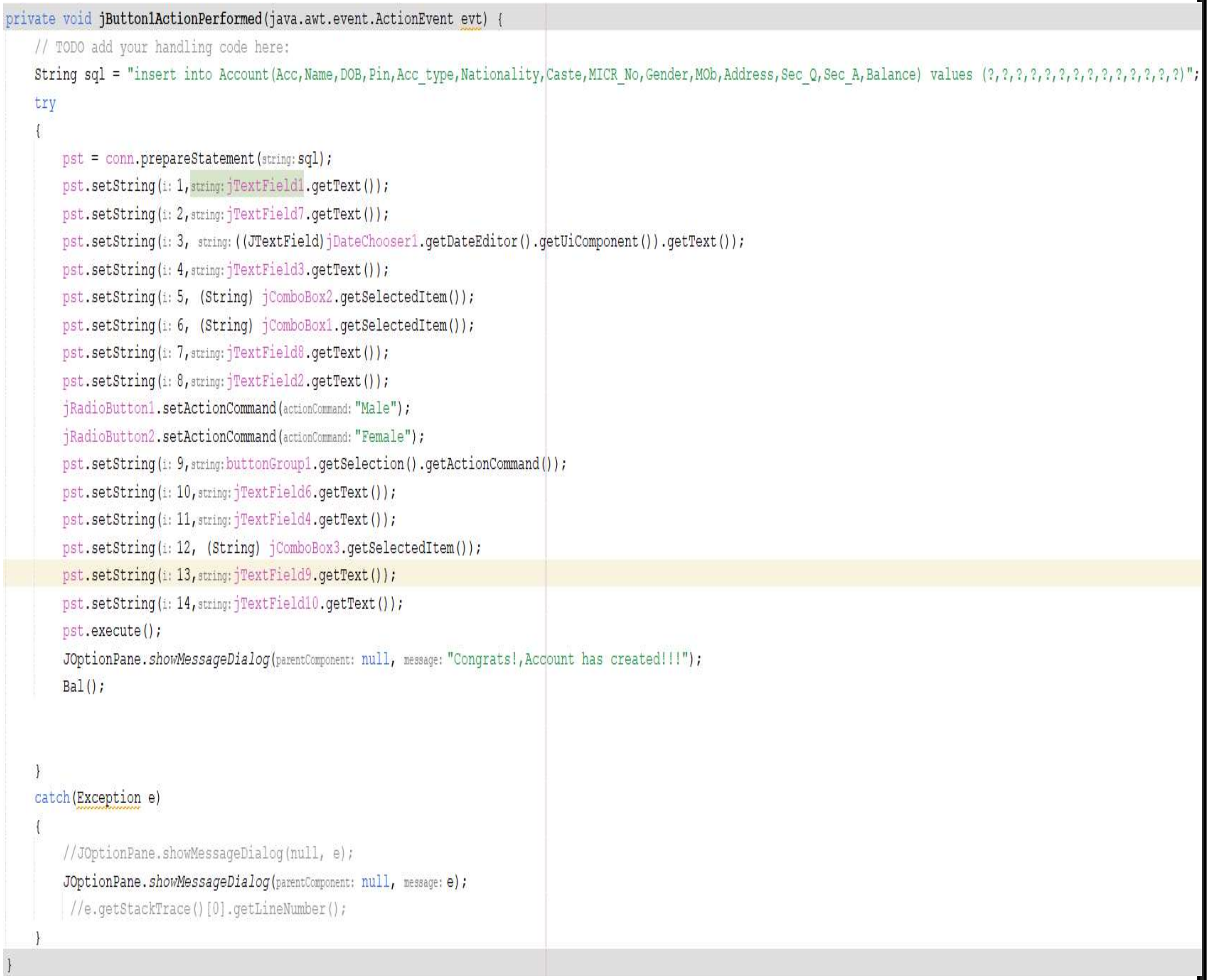
```
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.Random;
import javax.swing.JOptionPane;
import javax.swing.JTextField;

/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/
 * Click nbfs://nbhost/SystemFileSystem/Templates/
 */

/**
 *
 * @author MyPC
 */
public class Account extends javax.swing.JFrame {
Connection conn;
ResultSet rs;
PreparedStatement pst;
    /**
     * Creates new form Account
     */
    public Account() {
        super(title: "Create Account");
        initComponents();
        conn = javaconnect.ConnecrDb();
        //this.initComponents();
        RandomAcc();
        RandomMCIR();
        RandomPIN();
    }
```

#code for generating random number, pin, MICR No:



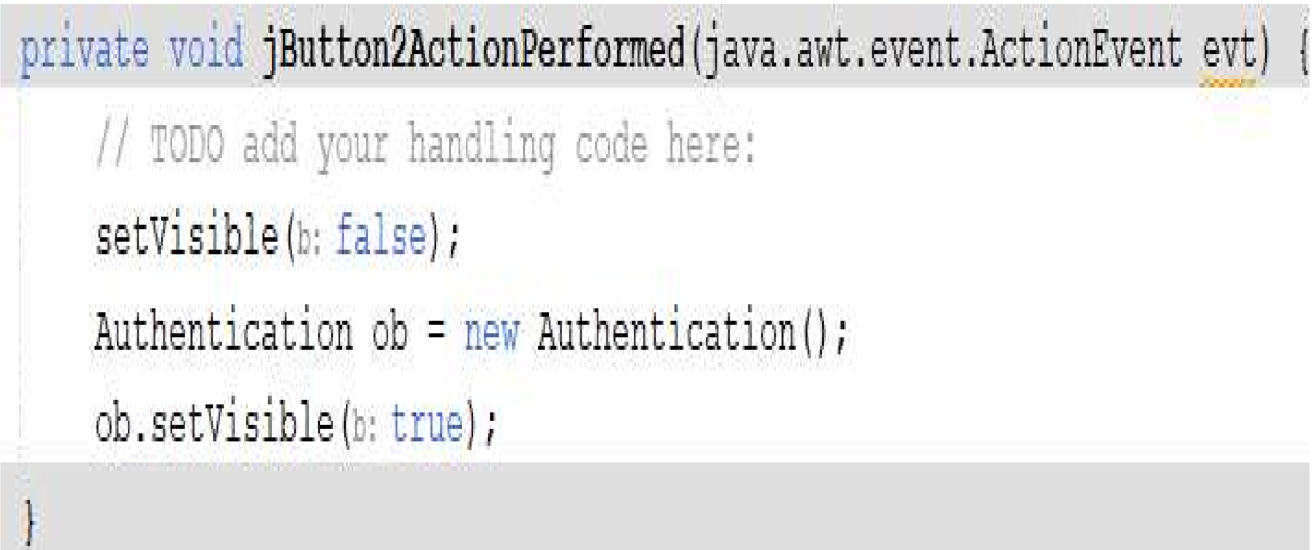
```
public void RandomAcc()
{
    Random ra = new Random();
    this.jTextField1.setText(""+ra.nextInt(bound: 10001));
}
public void RandomMCIR()
{
    Random ra = new Random();
    this.jTextField2.setText(""+ra.nextInt(bound: 10001));
}
public void RandomPIN()
{
    Random ra = new Random();
    this.jTextField3.setText(""+ra.nextInt(bound: 10001));
}
```

#Code for Save button in Account Page:



```java
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String sql = "insert into Account(Acc,Name,DOB,Pin,Acc_type,Nationality,Caste,MICR_No,Gender,MOb,Address,Sec_Q,Sec_A,Balance) values (?,?,?,?,?,?,?,?,?,?,?,?,?,?)";
    try
    {
        pst = conn.prepareStatement(string: sql);
        pst.setString(i: 1,string: jTextField1.getText());
        pst.setString(i: 2,string: jTextField7.getText());
        pst.setString(i: 3, string: ((JTextField)jDateChooser1.getDateEditor().getUiComponent()).getText());
        pst.setString(i: 4,string: jTextField3.getText());
        pst.setString(i: 5, (String) jComboBox2.getSelectedItem());
        pst.setString(i: 6, (String) jComboBox1.getSelectedItem());
        pst.setString(i: 7,string: jTextField8.getText());
        pst.setString(i: 8,string: jTextField2.getText());
        jRadioButton1.setActionCommand(actionCommand: "Male");
        jRadioButton2.setActionCommand(actionCommand: "Female");
        pst.setString(i: 9,string: buttonGroup1.getSelection().getActionCommand());
        pst.setString(i: 10,string: jTextField6.getText());
        pst.setString(i: 11,string: jTextField4.getText());
        pst.setString(i: 12, (String) jComboBox3.getSelectedItem());
        pst.setString(i: 13,string: jTextField9.getText());
        pst.setString(i: 14,string: jTextField10.getText());
        pst.execute();
        JOptionPane.showMessageDialog(parentComponent: null, message: "Congrats!,Account has created!!!");
        Bal();


    }
    catch(Exception e)
    {
        //JOptionPane.showMessageDialog(null, e);
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
         //e.getStackTrace()[0].getLineNumber();
    }
}
```

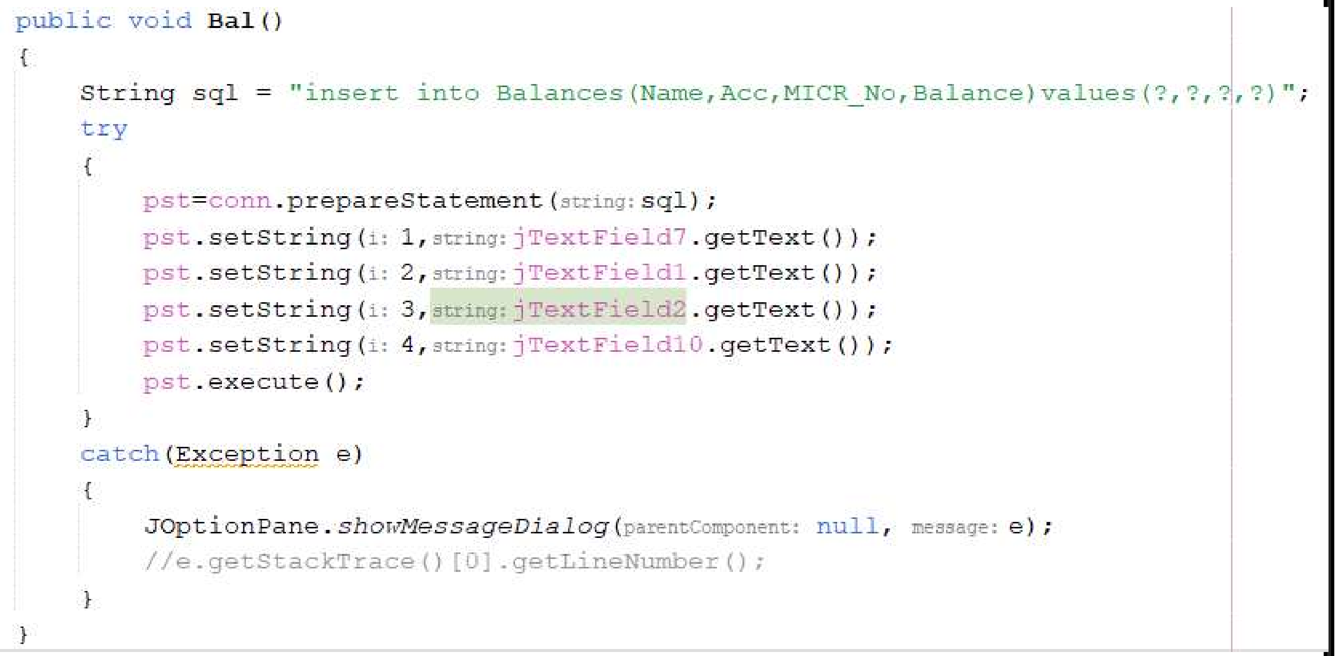
#code for Back button in Account Page:(Redrict from account to authentication page)



```java
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    setVisible(b: false);
    Authentication ob = new Authentication();
    ob.setVisible(b: true);
}
```

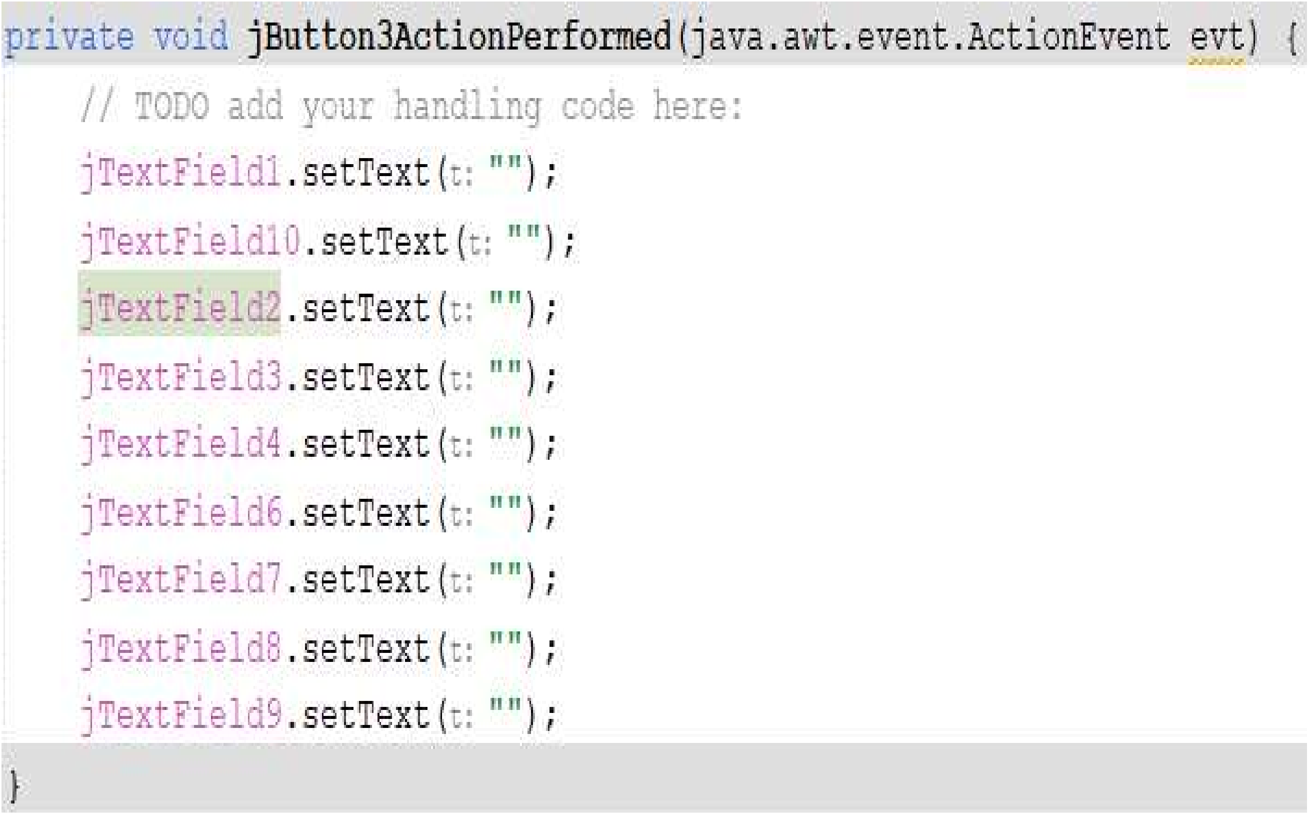
#Code for Bal() function which is called to store data into Balances table in DB

When data entered in account table,So thus called from account function definition:



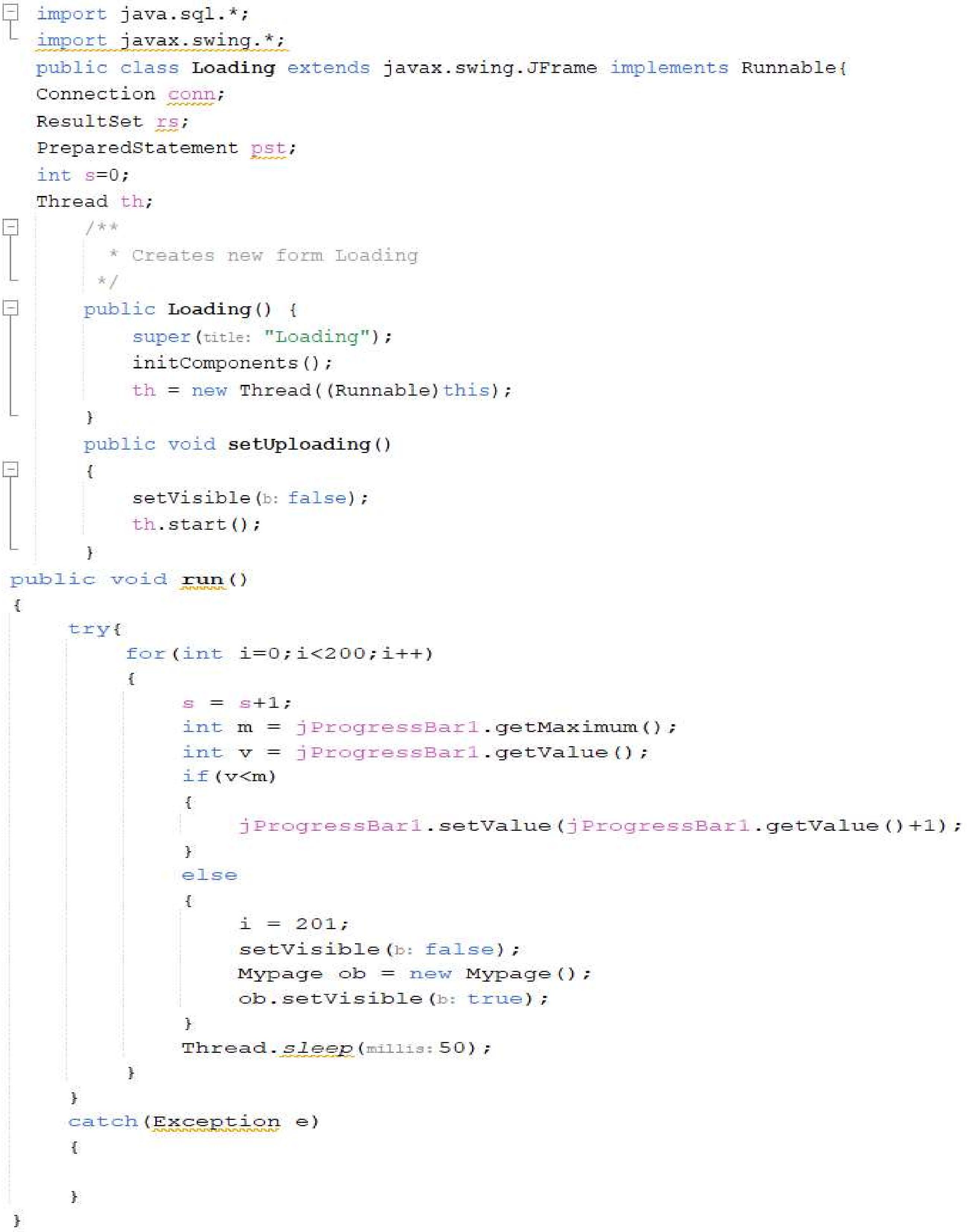
```
public void Bal()
{
    String sql = "insert into Balances(Name,Acc,MICR_No,Balance)values(?,?,?,?)";
    try
    {
        pst=conn.prepareStatement(string: sql);
        pst.setString(i: 1,string: jTextField7.getText());
        pst.setString(i: 2,string: jTextField1.getText());
        pst.setString(i: 3,string: jTextField2.getText());
        pst.setString(i: 4,string: jTextField10.getText());
        pst.execute();
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
        //e.getStackTrace()[0].getLineNumber();
    }
}
```

#Code for clear Button in Account Page:



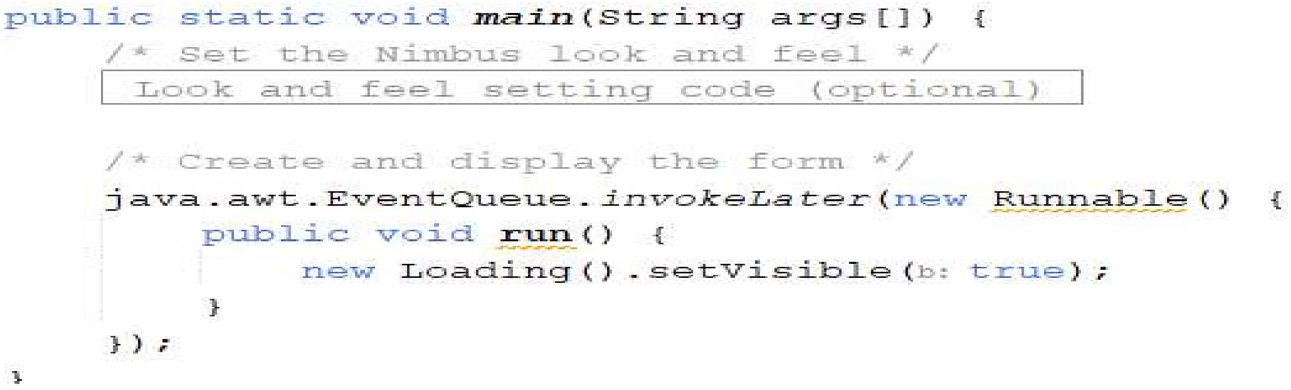
```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    jTextField1.setText(t: "");
    jTextField10.setText(t: "");
    jTextField2.setText(t: "");
    jTextField3.setText(t: "");
    jTextField4.setText(t: "");
    jTextField6.setText(t: "");
    jTextField7.setText(t: "");
    jTextField8.setText(t: "");
    jTextField9.setText(t: "");
}
```

#Code for Loading bar Page which appears after successful login from Authentication Page:



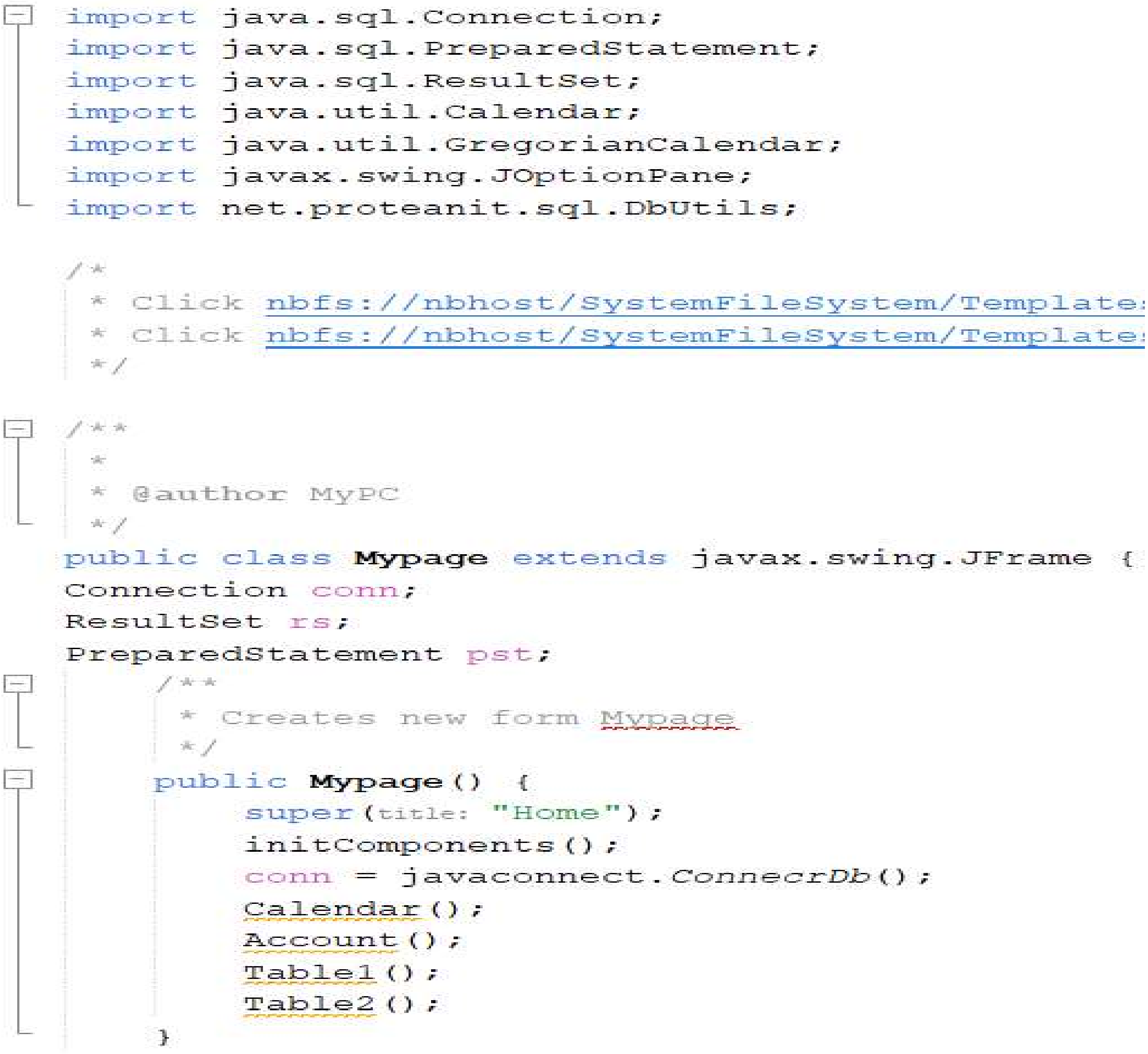
```
import java.sql.*;
import javax.swing.*;
public class Loading extends javax.swing.JFrame implements Runnable{
Connection conn;
ResultSet rs;
PreparedStatement pst;
int s=0;
Thread th;
    /**
     * Creates new form Loading
     */
    public Loading() {
        super(title: "Loading");
        initComponents();
        th = new Thread((Runnable)this);
    }
    public void setUploading()
    {
        setVisible(b: false);
        th.start();
    }
public void run()
{
    try{
        for(int i=0;i<200;i++)
        {
            s = s+1;
            int m = jProgressBar1.getMaximum();
            int v = jProgressBar1.getValue();
            if(v<m)
            {
                jProgressBar1.setValue(jProgressBar1.getValue()+1);
            }
            else
            {
                i = 201;
                setVisible(b: false);
                Mypage ob = new Mypage();
                ob.setVisible(b: true);
            }
            Thread.sleep(millis: 50);
        }
    }
    catch(Exception e)
    {

    }
}
```

#Main method for loading page:



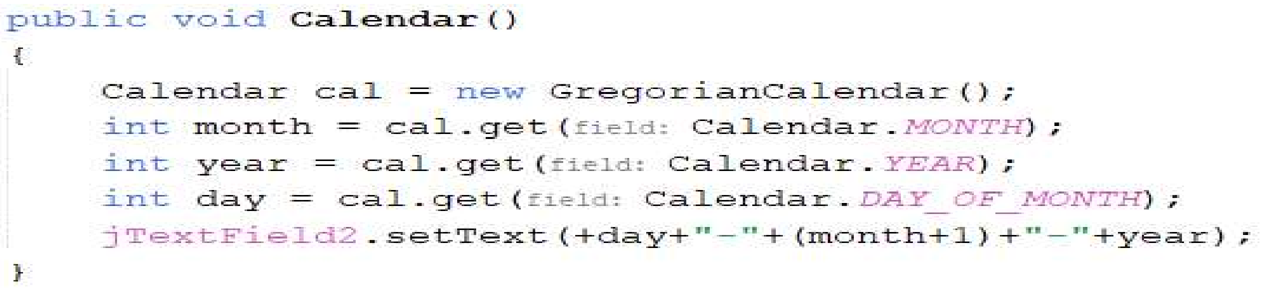
```
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    Look and feel setting code (optional)

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new Loading().setVisible(b: true);
        }
    });
}
```

#Code for My page(it is the main Page where we have all the bank functionalities):



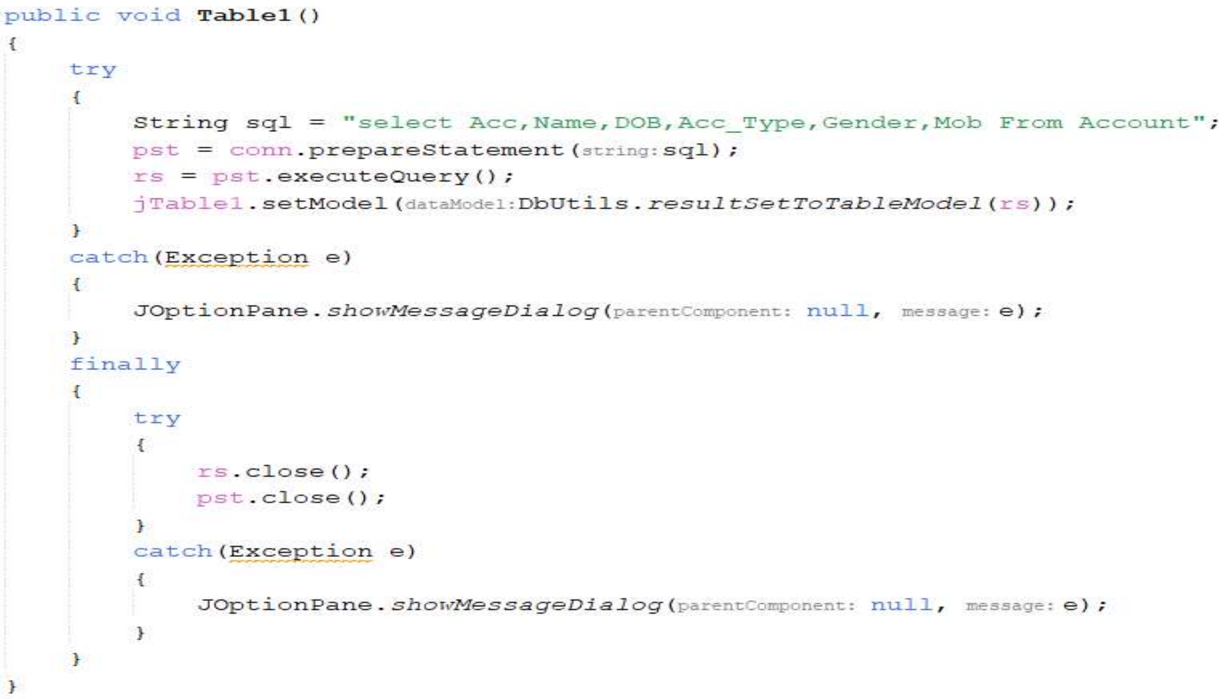
```
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.Calendar;
import java.util.GregorianCalendar;
import javax.swing.JOptionPane;
import net.proteanit.sql.DbUtils;

/*
 * Click nbfs://nbhost/SystemFileSystem/Template
 * Click nbfs://nbhost/SystemFileSystem/Template
 */

/**
 *
 * @author MyPC
 */
public class Mypage extends javax.swing.JFrame {
Connection conn;
ResultSet rs;
PreparedStatement pst;
    /**
     * Creates new form Mypage
     */
    public Mypage() {
        super(title: "Home");
        initComponents();
        conn = javaconnect.ConnecrDb();
        Calendar();
        Account();
        Table1();
        Table2();
    }
```

#code for automatic calendar for representing Present Date in My page:

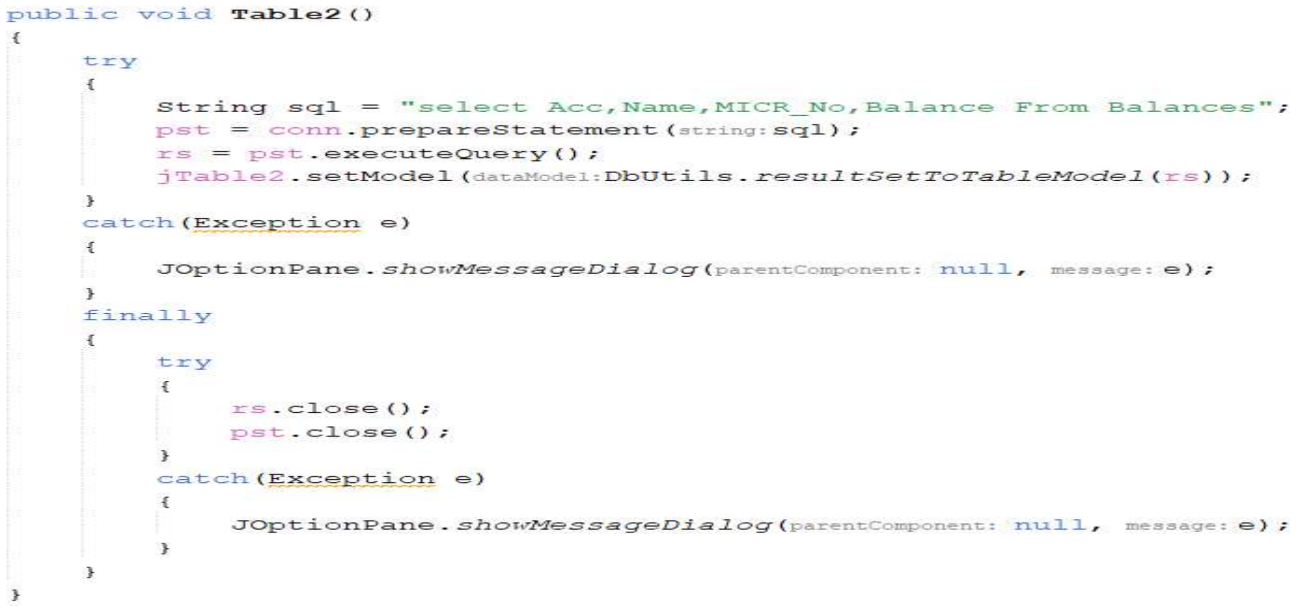


```
public void Calendar()
{
     Calendar cal = new GregorianCalendar();
     int month = cal.get(field: Calendar.MONTH);
     int year = cal.get(field: Calendar.YEAR);
     int day = cal.get(field: Calendar.DAY_OF_MONTH);
     jTextField2.setText(+day+"-"+(month+1)+"-"+year);
}
```

#code for Tables which show Transaction history & total number of Accounts:

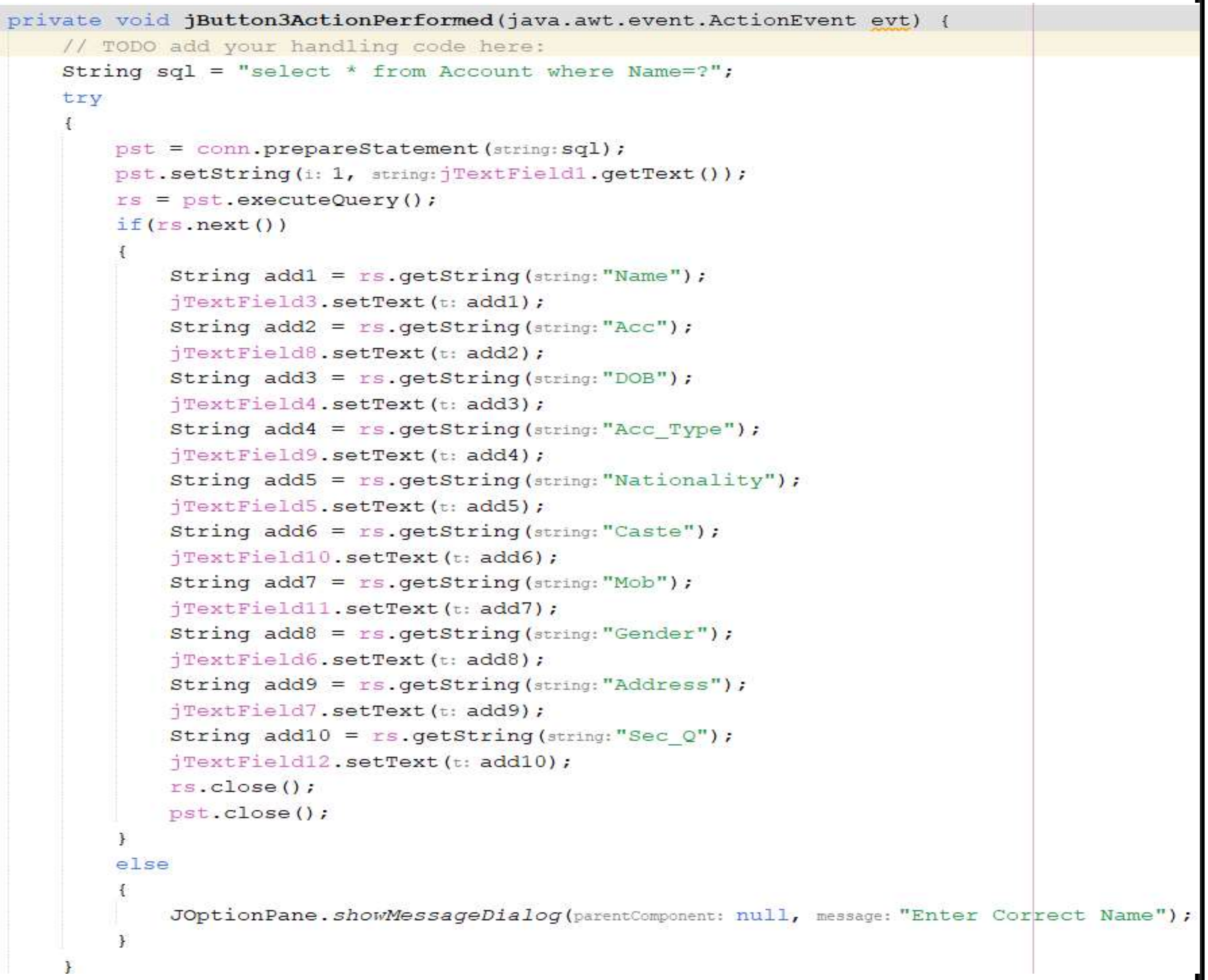


```
public void Table1()
{
    try
    {
        String sql = "select Acc,Name,DOB,Acc_Type,Gender,Mob From Account";
        pst = conn.prepareStatement(string:sql);
        rs = pst.executeQuery();
        jTable1.setModel(dataModel:DbUtils.resultSetToTableModel(rs));
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
    finally
    {
        try
        {
            rs.close();
            pst.close();
        }
        catch(Exception e)
        {
            JOptionPane.showMessageDialog(parentComponent: null, message: e);
        }
    }
}
```

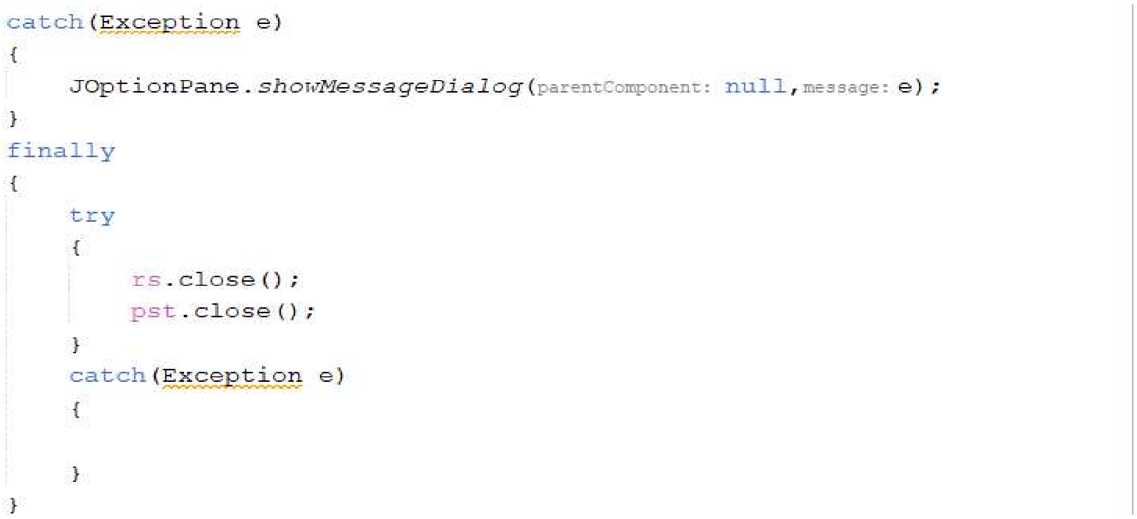


```
public void Table2()
{
    try
    {
        String sql = "select Acc,Name,MICR_No,Balance From Balances";
        pst = conn.prepareStatement(string:sql);
        rs = pst.executeQuery();
        jTable2.setModel(dataModel:DbUtils.resultSetToTableModel(rs));
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
    finally
    {
        try
        {
            rs.close();
            pst.close();
        }
        catch(Exception e)
        {
            JOptionPane.showMessageDialog(parentComponent: null, message: e);
        }
    }
}
```

Code for Eye button which shows Account Details using User Name:

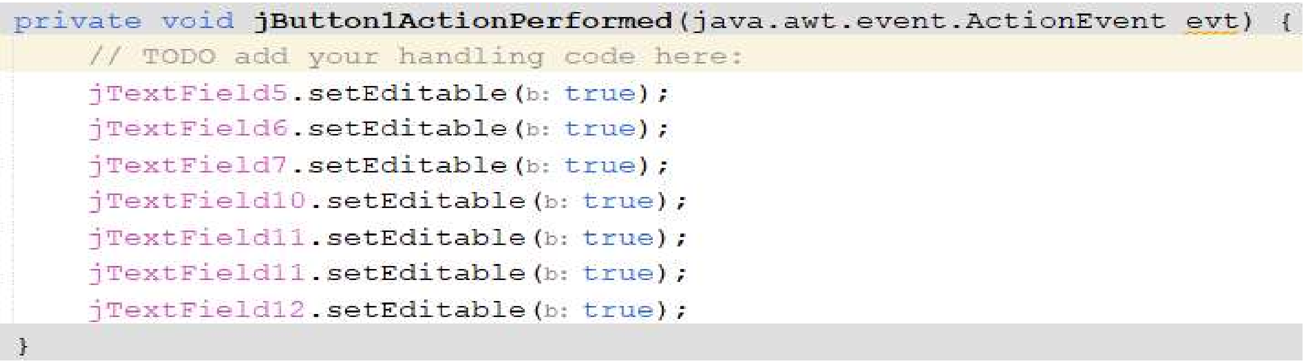


```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String sql = "select * from Account where Name=?";
    try
    {
        pst = conn.prepareStatement(string:sql);
        pst.setString(i: 1, string:jTextField1.getText());
        rs = pst.executeQuery();
        if(rs.next())
        {
            String add1 = rs.getString(string:"Name");
            jTextField3.setText(t: add1);
            String add2 = rs.getString(string:"Acc");
            jTextField8.setText(t: add2);
            String add3 = rs.getString(string:"DOB");
            jTextField4.setText(t: add3);
            String add4 = rs.getString(string:"Acc_Type");
            jTextField9.setText(t: add4);
            String add5 = rs.getString(string:"Nationality");
            jTextField5.setText(t: add5);
            String add6 = rs.getString(string:"Caste");
            jTextField10.setText(t: add6);
            String add7 = rs.getString(string:"Mob");
            jTextField11.setText(t: add7);
            String add8 = rs.getString(string:"Gender");
            jTextField6.setText(t: add8);
            String add9 = rs.getString(string:"Address");
            jTextField7.setText(t: add9);
            String add10 = rs.getString(string:"Sec_Q");
            jTextField12.setText(t: add10);
            rs.close();
            pst.close();
        }
        else
        {
            JOptionPane.showMessageDialog(parentComponent: null, message: "Enter Correct Name");
        }
    }
```



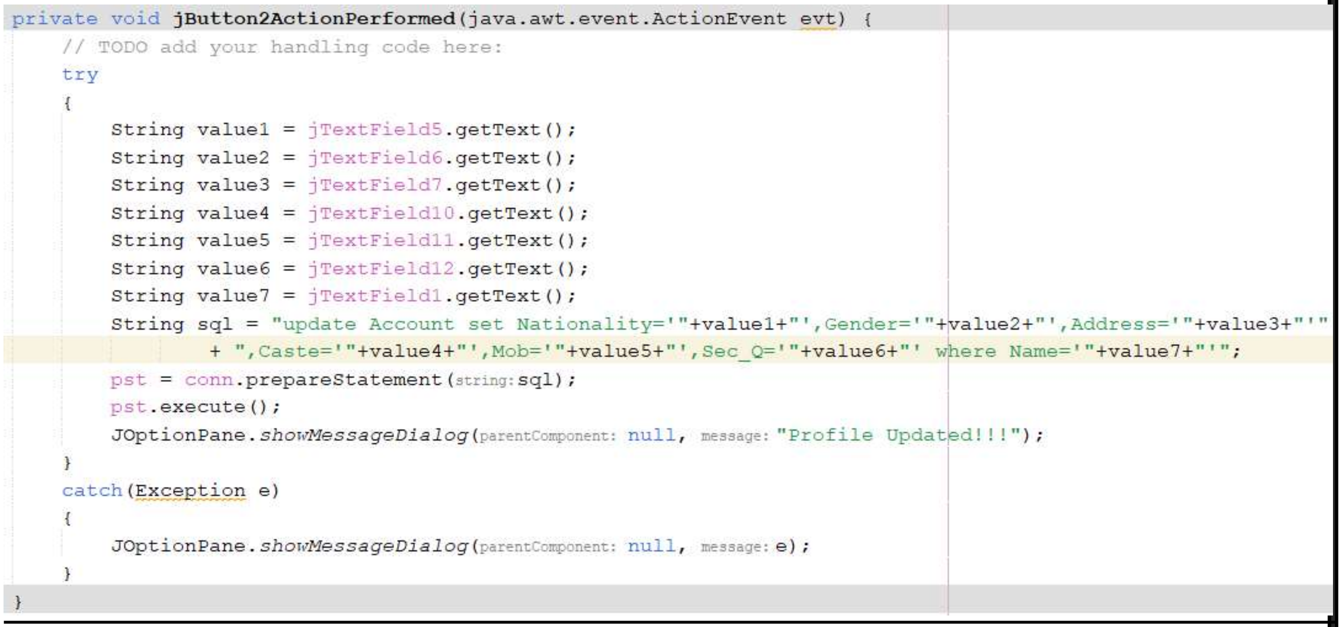
```
catch(Exception e)
{
    JOptionPane.showMessageDialog(parentComponent: null,message: e);
}
finally
{
    try
    {
        rs.close();
        pst.close();
    }
    catch(Exception e)
    {

    }
}
```

#Code for Edit button in Profile tab(JtabedPane) of Mypage:



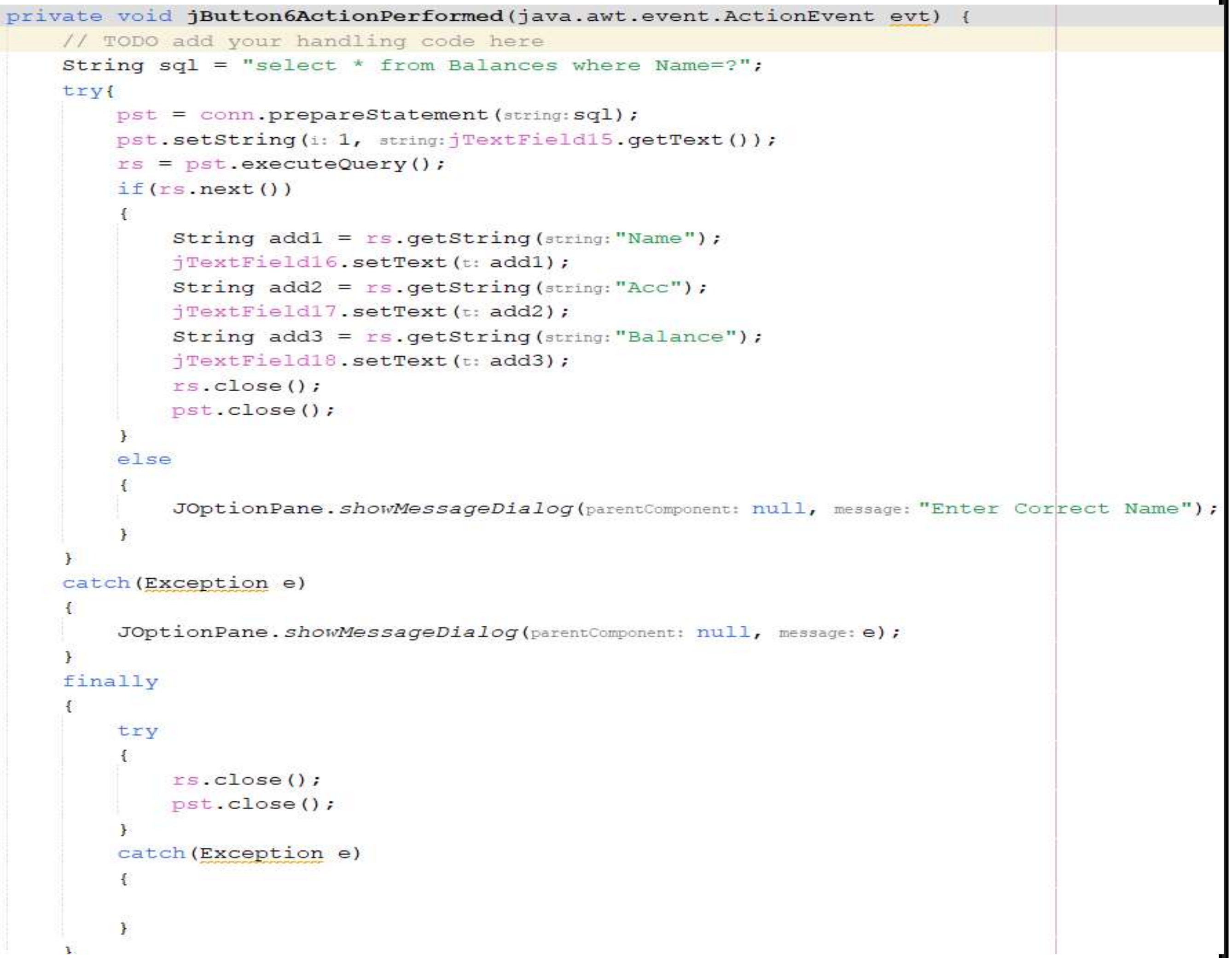
```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    jTextField5.setEditable(b: true);
    jTextField6.setEditable(b: true);
    jTextField7.setEditable(b: true);
    jTextField10.setEditable(b: true);
    jTextField11.setEditable(b: true);
    jTextField11.setEditable(b: true);
    jTextField12.setEditable(b: true);
}
```

#Code for Save button in Profile tab(JtabedPane) of Mypage:



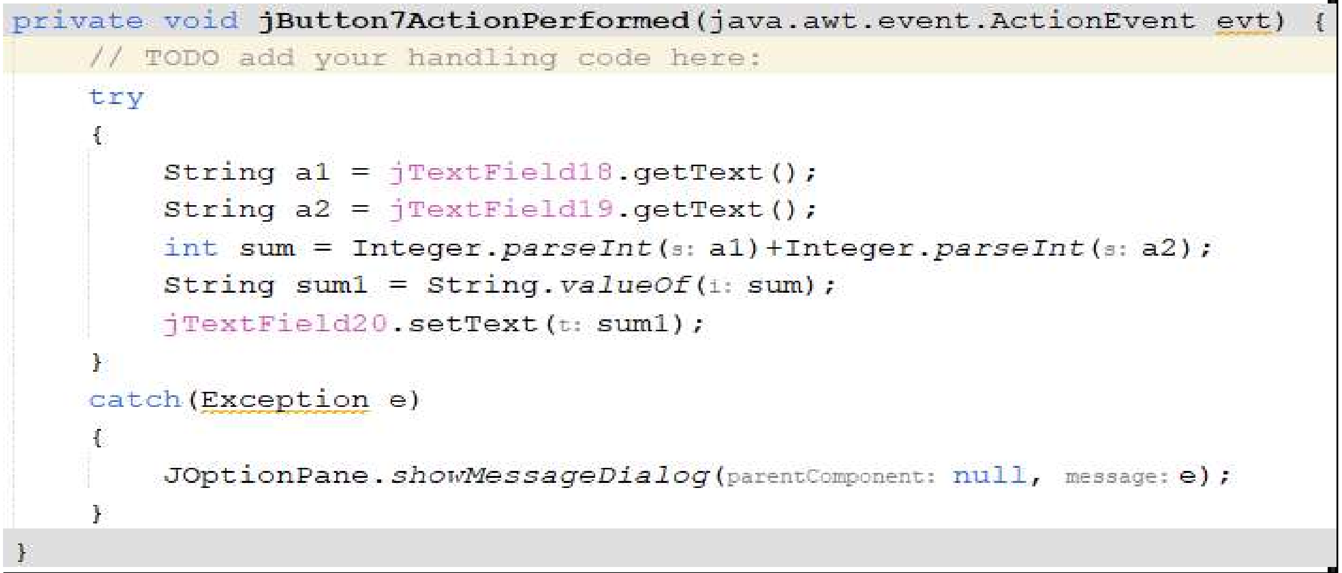
```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try
    {
        String value1 = jTextField5.getText();
        String value2 = jTextField6.getText();
        String value3 = jTextField7.getText();
        String value4 = jTextField10.getText();
        String value5 = jTextField11.getText();
        String value6 = jTextField12.getText();
        String value7 = jTextField1.getText();
        String sql = "update Account set Nationality='"+value1+"',Gender='"+value2+"',Address='"+value3+"'"
                + ",Caste='"+value4+"',Mob='"+value5+"',Sec_Q='"+value6+"' where Name='"+value7+"'";
        pst = conn.prepareStatement(string: sql);
        pst.execute();
        JOptionPane.showMessageDialog(parentComponent: null, message: "Profile Updated!!!");
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Search button in Deposit tab of Mypage:



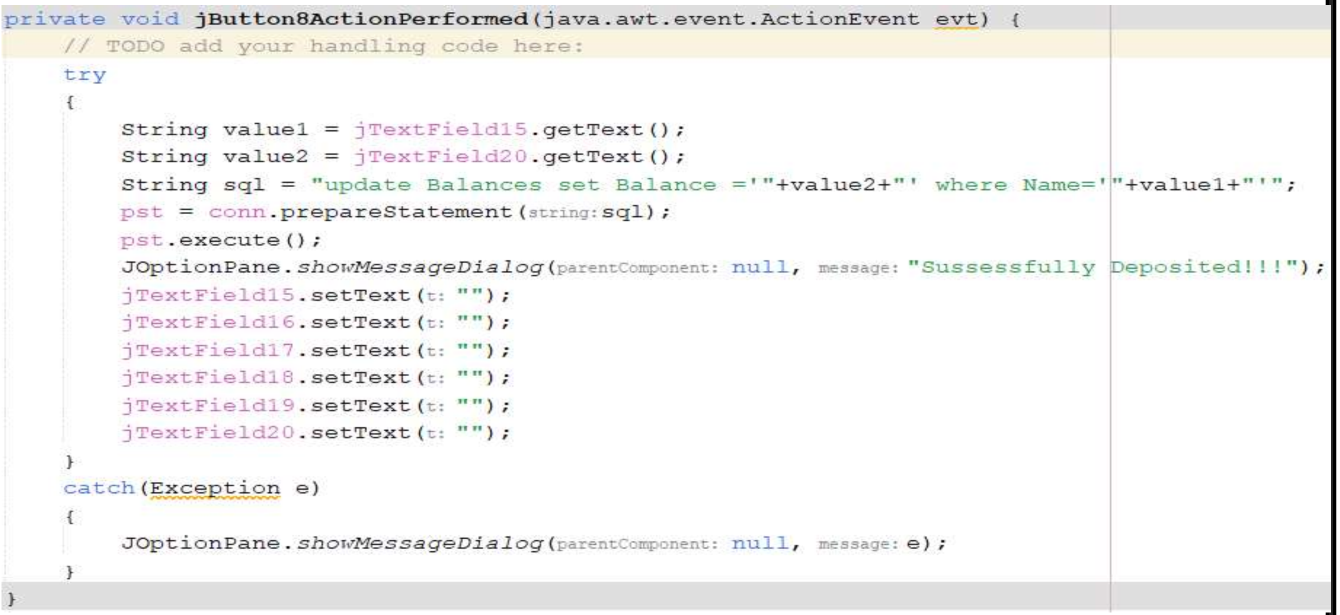
```
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here
    String sql = "select * from Balances where Name=?";
    try{
        pst = conn.prepareStatement(string: sql);
        pst.setString(i: 1, string: jTextField15.getText());
        rs = pst.executeQuery();
        if(rs.next())
        {
            String add1 = rs.getString(string: "Name");
            jTextField16.setText(t: add1);
            String add2 = rs.getString(string: "Acc");
            jTextField17.setText(t: add2);
            String add3 = rs.getString(string: "Balance");
            jTextField18.setText(t: add3);
            rs.close();
            pst.close();
        }
        else
        {
            JOptionPane.showMessageDialog(parentComponent: null, message: "Enter Correct Name");
        }
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
    finally
    {
        try
        {
            rs.close();
            pst.close();
        }
        catch(Exception e)
        {

        }
    }
```

#Code for Total button in Deposit tab of Mypage:



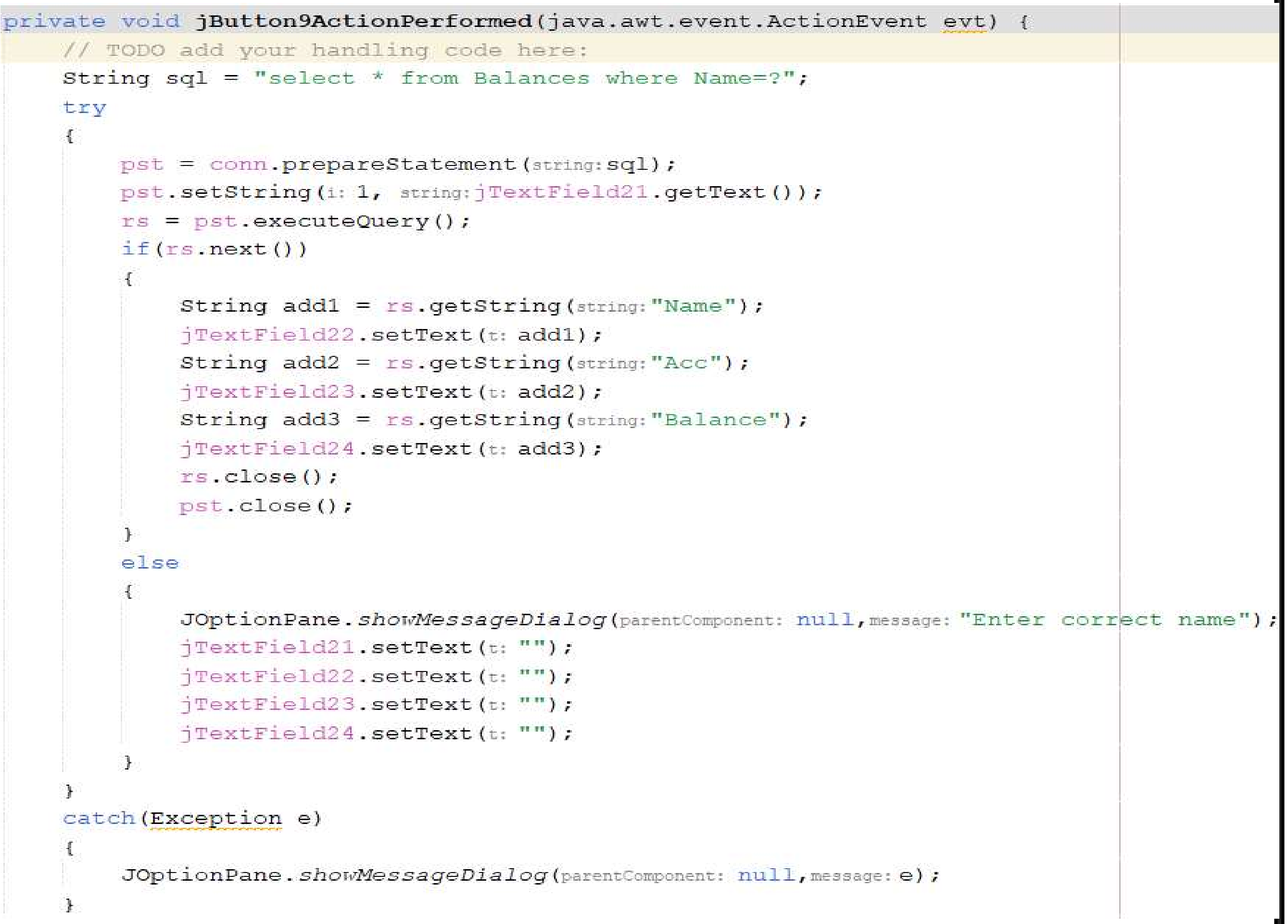
```
private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try
    {
        String a1 = jTextField18.getText();
        String a2 = jTextField19.getText();
        int sum = Integer.parseInt(s: a1)+Integer.parseInt(s: a2);
        String sum1 = String.valueOf(i: sum);
        jTextField20.setText(t: sum1);
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Deposit button in Deposit tab of Mypage:



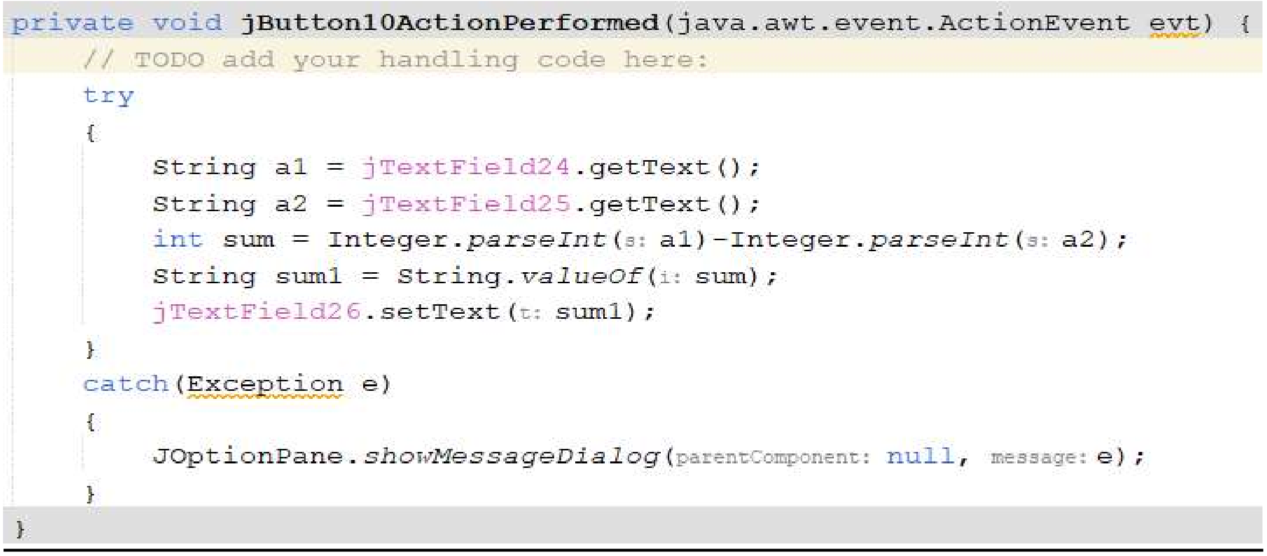
```
private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try
    {
        String value1 = jTextField15.getText();
        String value2 = jTextField20.getText();
        String sql = "update Balances set Balance ='"+value2+"' where Name='"+value1+"'";
        pst = conn.prepareStatement(string:sql);
        pst.execute();
        JOptionPane.showMessageDialog(parentComponent: null, message: "Sussessfully Deposited!!!");
        jTextField15.setText(t: "");
        jTextField16.setText(t: "");
        jTextField17.setText(t: "");
        jTextField18.setText(t: "");
        jTextField19.setText(t: "");
        jTextField20.setText(t: "");
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Search button in Transfer tab of Mypage:



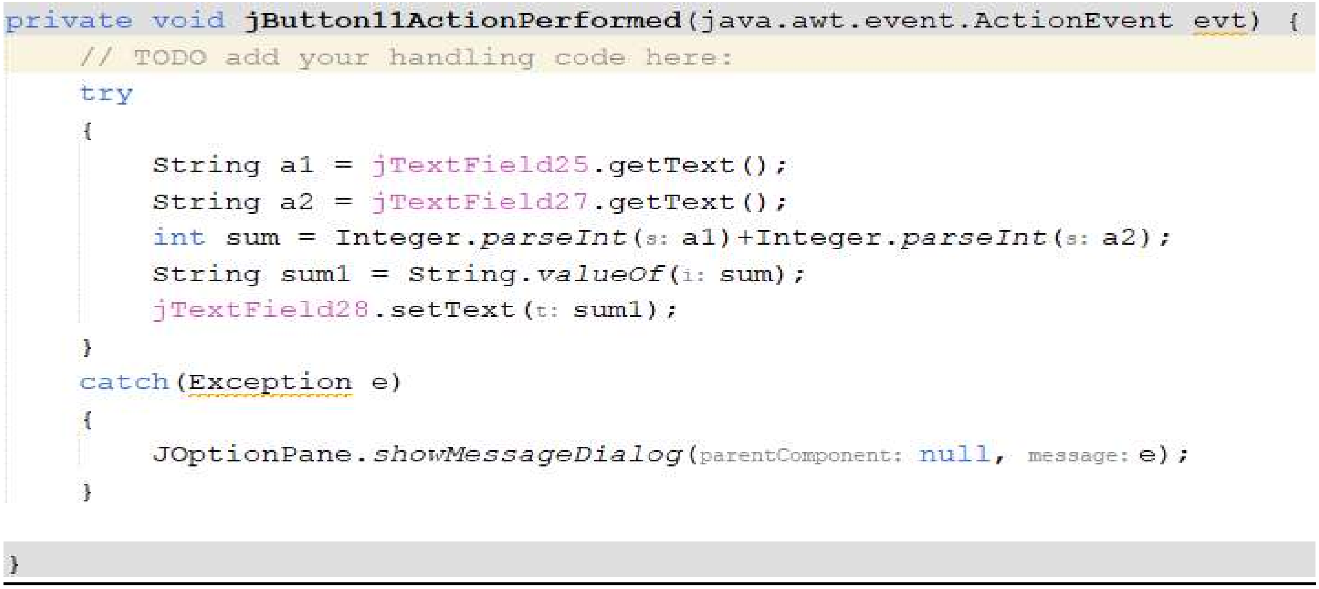
```
private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String sql = "select * from Balances where Name=?";
    try
    {
        pst = conn.prepareStatement(string:sql);
        pst.setString(i: 1, string:jTextField21.getText());
        rs = pst.executeQuery();
        if(rs.next())
        {
            String add1 = rs.getString(string:"Name");
            jTextField22.setText(t: add1);
            String add2 = rs.getString(string:"Acc");
            jTextField23.setText(t: add2);
            String add3 = rs.getString(string:"Balance");
            jTextField24.setText(t: add3);
            rs.close();
            pst.close();
        }
        else
        {
            JOptionPane.showMessageDialog(parentComponent: null,message: "Enter correct name");
            jTextField21.setText(t: "");
            jTextField22.setText(t: "");
            jTextField23.setText(t: "");
            jTextField24.setText(t: "");
        }
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null,message: e);
    }
```

#Code for Total button in Transfer tab of Mypage:



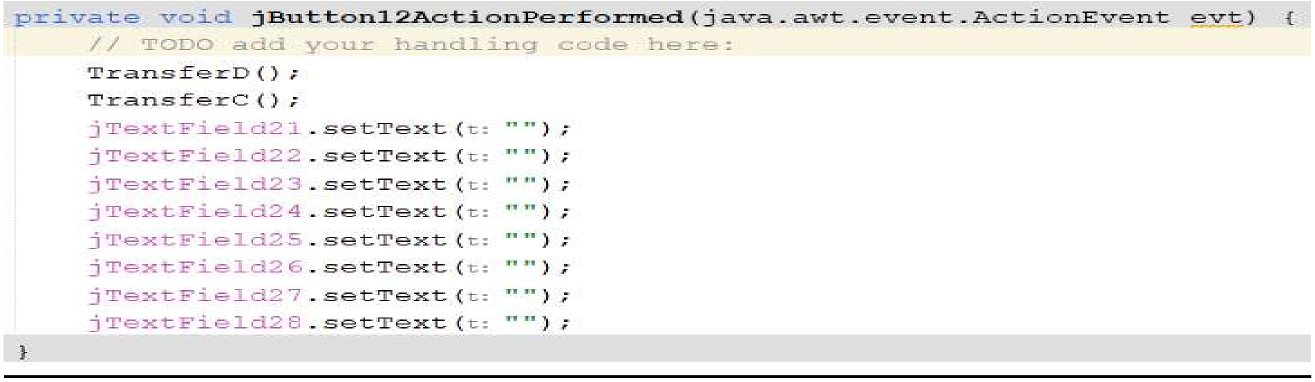
```
private void jButton10ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try
    {
        String a1 = jTextField24.getText();
        String a2 = jTextField25.getText();
        int sum = Integer.parseInt(s: a1)-Integer.parseInt(s: a2);
        String sum1 = String.valueOf(i: sum);
        jTextField26.setText(t: sum1);
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Show button in Transfer tab of Mypage:



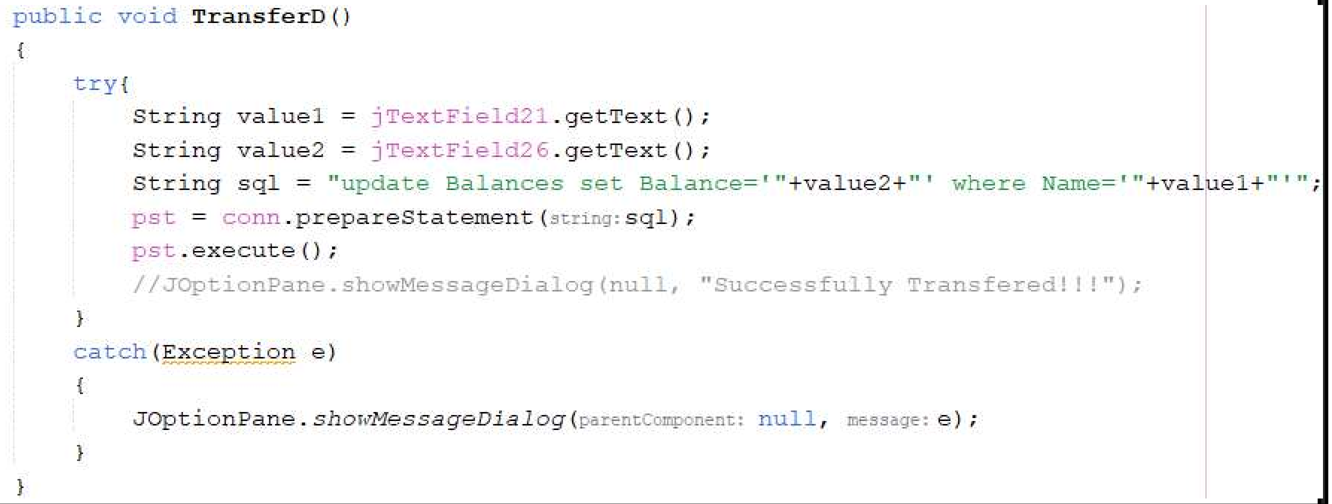
```
private void jButton11ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try
    {
        String a1 = jTextField25.getText();
        String a2 = jTextField27.getText();
        int sum = Integer.parseInt(s: a1)+Integer.parseInt(s: a2);
        String sum1 = String.valueOf(i: sum);
        jTextField28.setText(t: sum1);
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }

}
```

#Code for Transfer button in Transfer tab of Mypage:



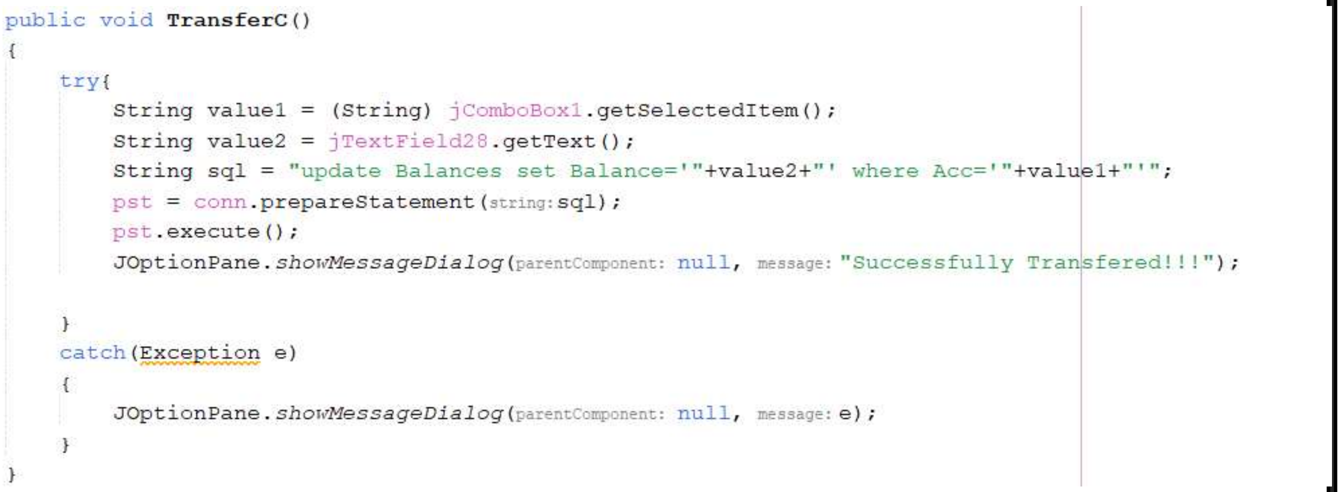
```
private void jButton12ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    TransferD();
    TransferC();
    jTextField21.setText(t: "");
    jTextField22.setText(t: "");
    jTextField23.setText(t: "");
    jTextField24.setText(t: "");
    jTextField25.setText(t: "");
    jTextField26.setText(t: "");
    jTextField27.setText(t: "");
    jTextField28.setText(t: "");
}
```

#Code for Tansfer_D() function:



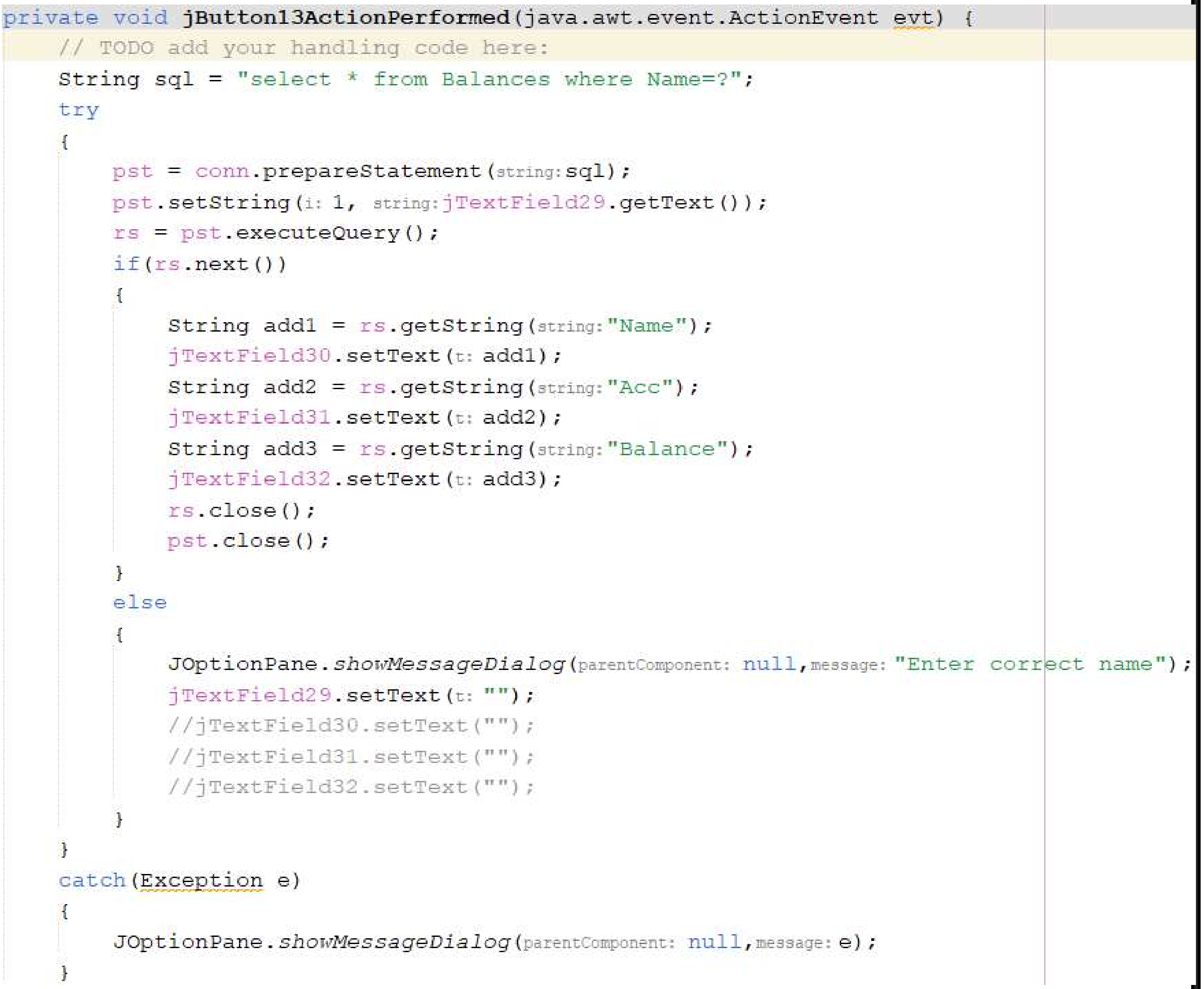
```
public void TransferD()
{
    try{
        String value1 = jTextField21.getText();
        String value2 = jTextField26.getText();
        String sql = "update Balances set Balance='"+value2+"' where Name='"+value1+"'";
        pst = conn.prepareStatement(string:sql);
        pst.execute();
        //JOptionPane.showMessageDialog(null, "Successfully Transfered!!!");
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Tansfer_C() function:



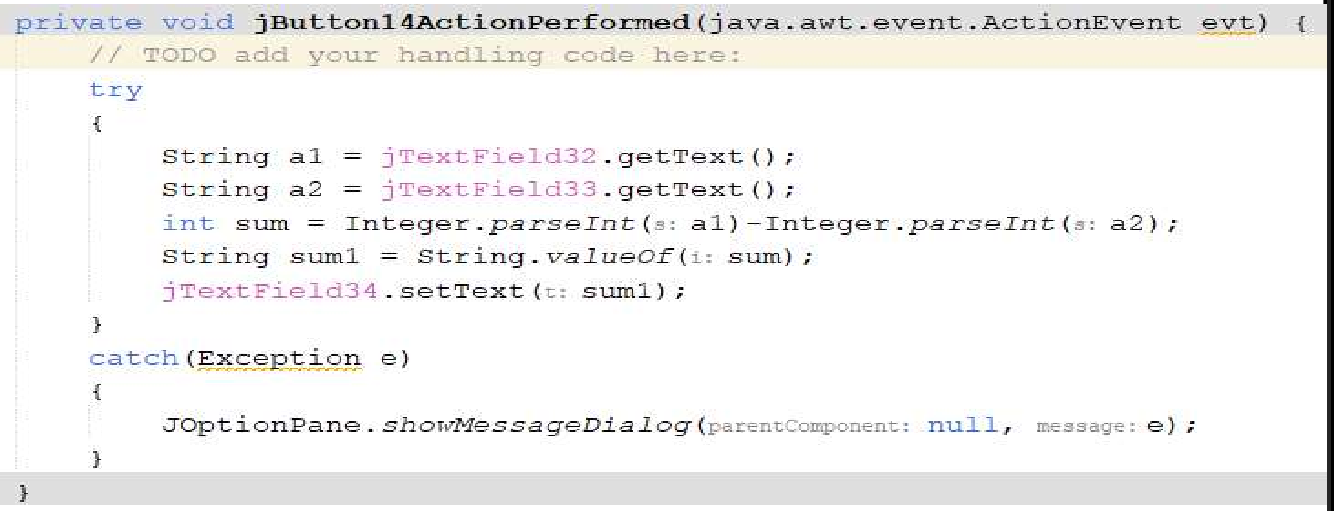
```
public void TransferC()
{
    try{
        String value1 = (String) jComboBox1.getSelectedItem();
        String value2 = jTextField28.getText();
        String sql = "update Balances set Balance='"+value2+"' where Acc='"+value1+"'";
        pst = conn.prepareStatement(string:sql);
        pst.execute();
        JOptionPane.showMessageDialog(parentComponent: null, message: "Successfully Transfered!!!");

    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Search button in Withdrawl tab of Mypage:



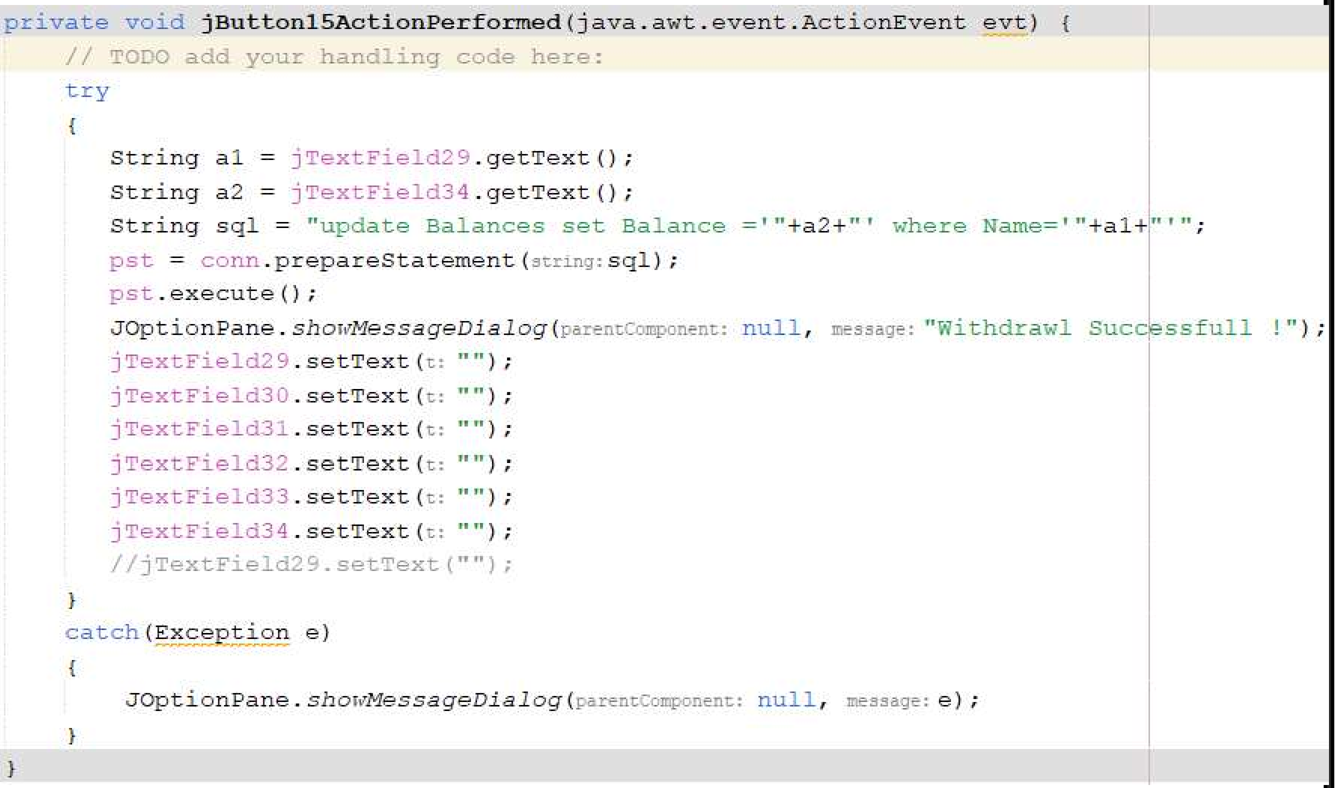
```
private void jButton13ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String sql = "select * from Balances where Name=?";
    try
    {
        pst = conn.prepareStatement(string:sql);
        pst.setString(i: 1, string:jTextField29.getText());
        rs = pst.executeQuery();
        if(rs.next())
        {
            String add1 = rs.getString(string:"Name");
            jTextField30.setText(t: add1);
            String add2 = rs.getString(string:"Acc");
            jTextField31.setText(t: add2);
            String add3 = rs.getString(string:"Balance");
            jTextField32.setText(t: add3);
            rs.close();
            pst.close();
        }
        else
        {
            JOptionPane.showMessageDialog(parentComponent: null,message: "Enter correct name");
            jTextField29.setText(t: "");
            //jTextField30.setText("");
            //jTextField31.setText("");
            //jTextField32.setText("");
        }
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null,message: e);
    }
```

#Code for Show button in Withdrawl tab of Mypage:



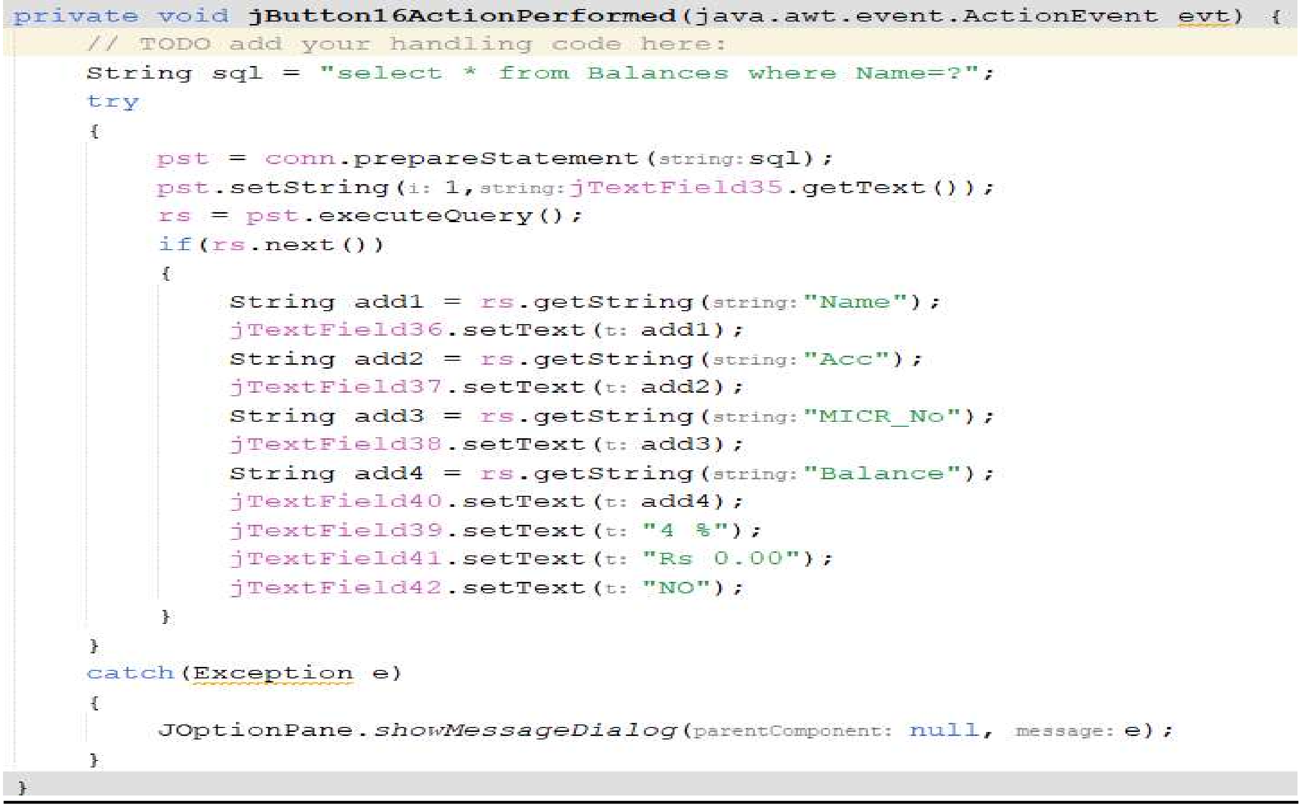
```
private void jButton14ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try
    {
        String a1 = jTextField32.getText();
        String a2 = jTextField33.getText();
        int sum = Integer.parseInt(s: a1)-Integer.parseInt(s: a2);
        String sum1 = String.valueOf(i: sum);
        jTextField34.setText(t: sum1);
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Withdraw button in Withdrawl tab of Mypage:



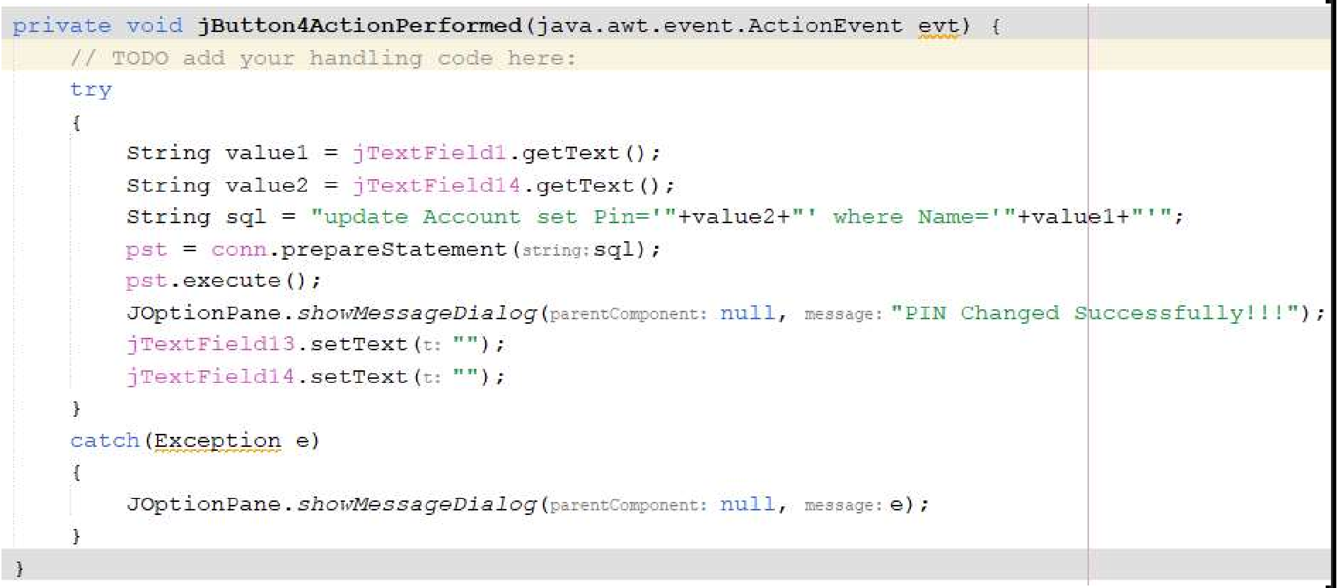
```
private void jButton15ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try
    {
       String a1 = jTextField29.getText();
       String a2 = jTextField34.getText();
       String sql = "update Balances set Balance ='"+a2+"' where Name='"+a1+"'";
       pst = conn.prepareStatement(string:sql);
       pst.execute();
       JOptionPane.showMessageDialog(parentComponent: null, message: "Withdrawl Successfull !");
       jTextField29.setText(t: "");
       jTextField30.setText(t: "");
       jTextField31.setText(t: "");
       jTextField32.setText(t: "");
       jTextField33.setText(t: "");
       jTextField34.setText(t: "");
       //jTextField29.setText("");
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Search button in View Balance tab of Mypage:



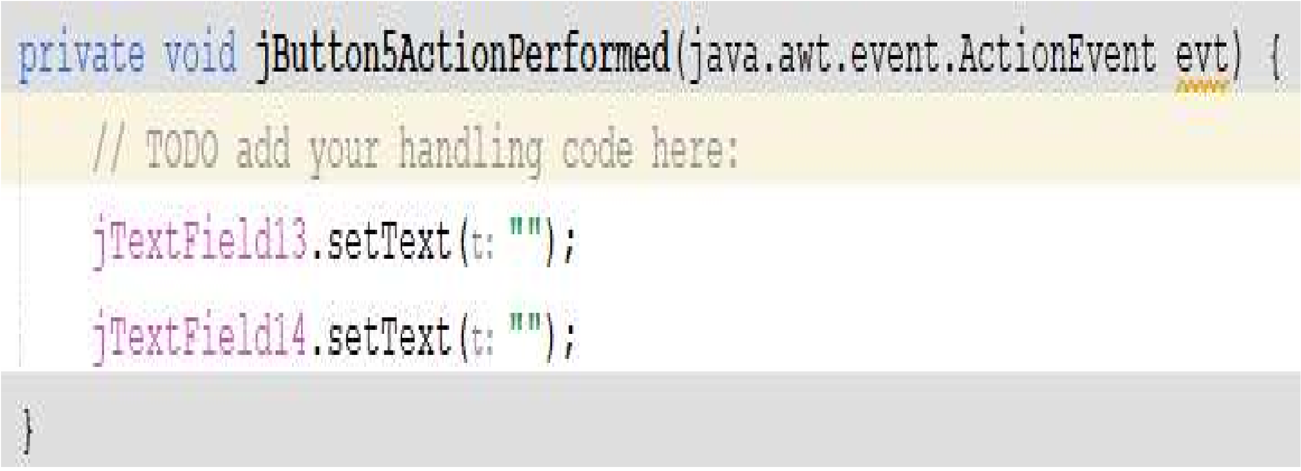
```
private void jButton16ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String sql = "select * from Balances where Name=?";
    try
    {
        pst = conn.prepareStatement(string:sql);
        pst.setString(i: 1,string:jTextField35.getText());
        rs = pst.executeQuery();
        if(rs.next())
        {
            String add1 = rs.getString(string:"Name");
            jTextField36.setText(t: add1);
            String add2 = rs.getString(string:"Acc");
            jTextField37.setText(t: add2);
            String add3 = rs.getString(string:"MICR_No");
            jTextField38.setText(t: add3);
            String add4 = rs.getString(string:"Balance");
            jTextField40.setText(t: add4);
            jTextField39.setText(t: "4 %");
            jTextField41.setText(t: "Rs 0.00");
            jTextField42.setText(t: "NO");
        }
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Change button in ChangePin tab of Mypage:



```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try
    {
        String value1 = jTextField1.getText();
        String value2 = jTextField14.getText();
        String sql = "update Account set Pin='"+value2+"' where Name='"+value1+"'";
        pst = conn.prepareStatement(string:sql);
        pst.execute();
        JOptionPane.showMessageDialog(parentComponent: null, message: "PIN Changed Successfully!!!");
        jTextField13.setText(t: "");
        jTextField14.setText(t: "");
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(parentComponent: null, message: e);
    }
}
```

#Code for Clear button in ChangePin tab of Mypage:



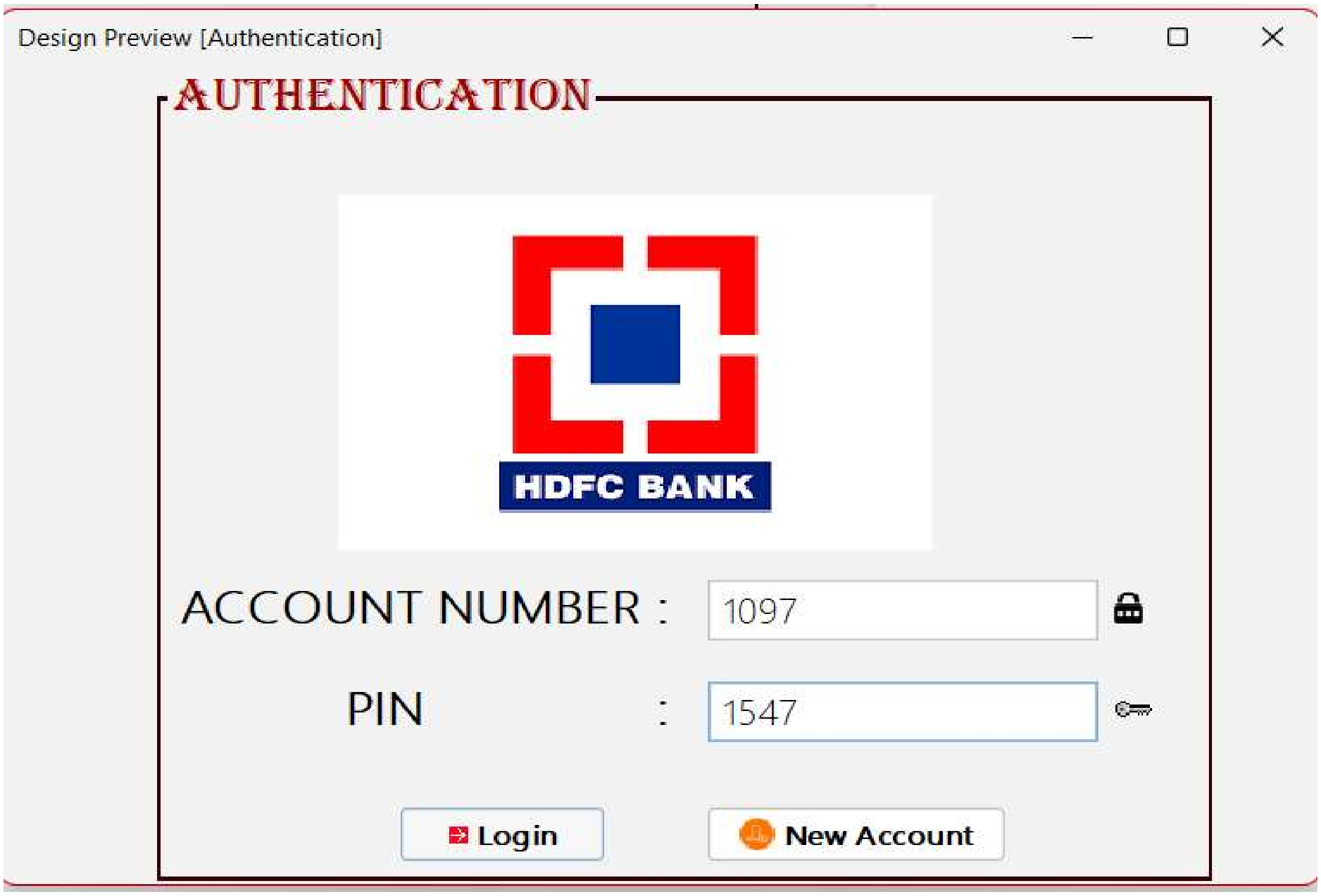
```
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    jTextField13.setText(t: "");
    jTextField14.setText(t: "");
}
```

CHAPTER: 5

RESULTS

OUTPUT:

#Authentication Page:



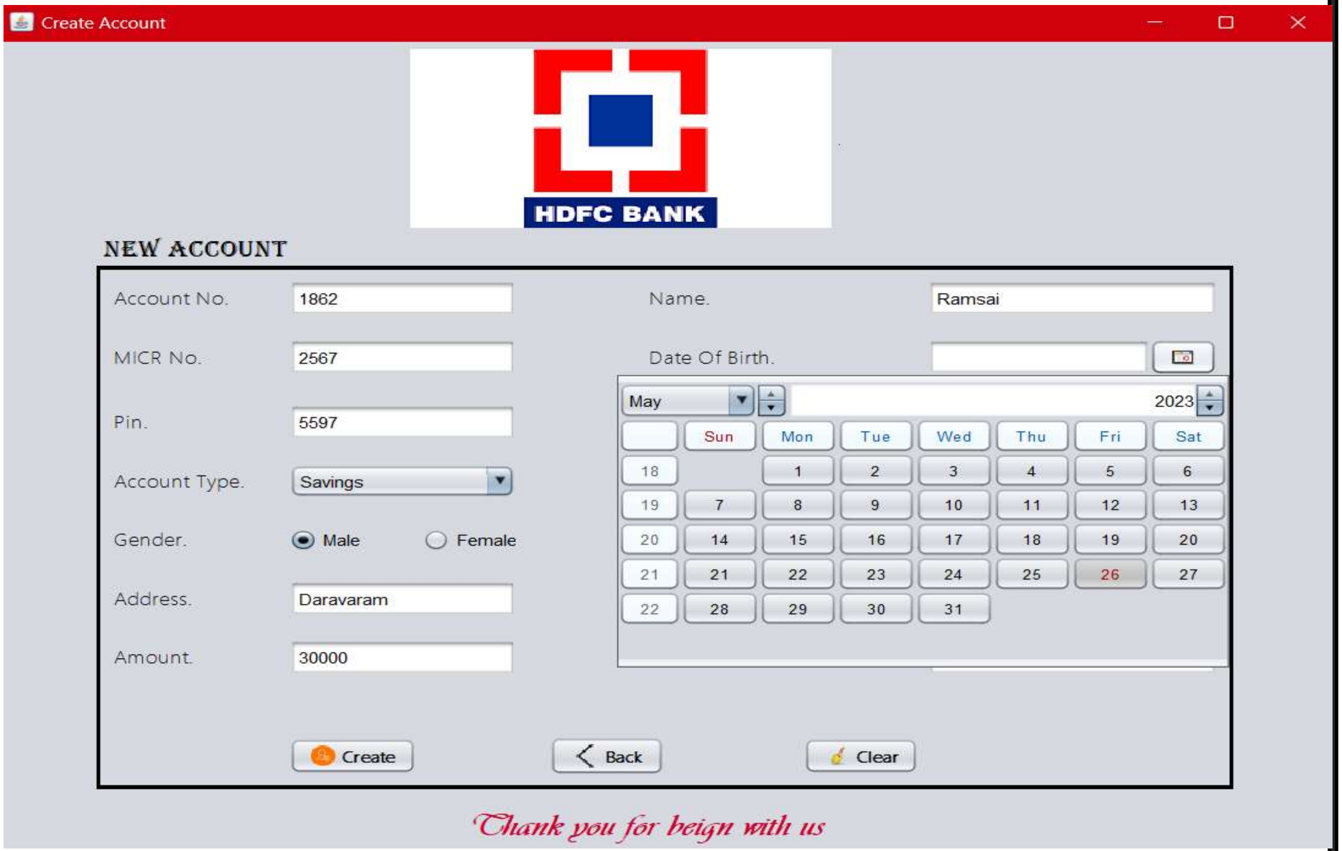
Design Preview [Authentication]

AUTHENTICATION

ACCOUNT NUMBER : 1097

PIN : 1547

#Account Page:



Create Account

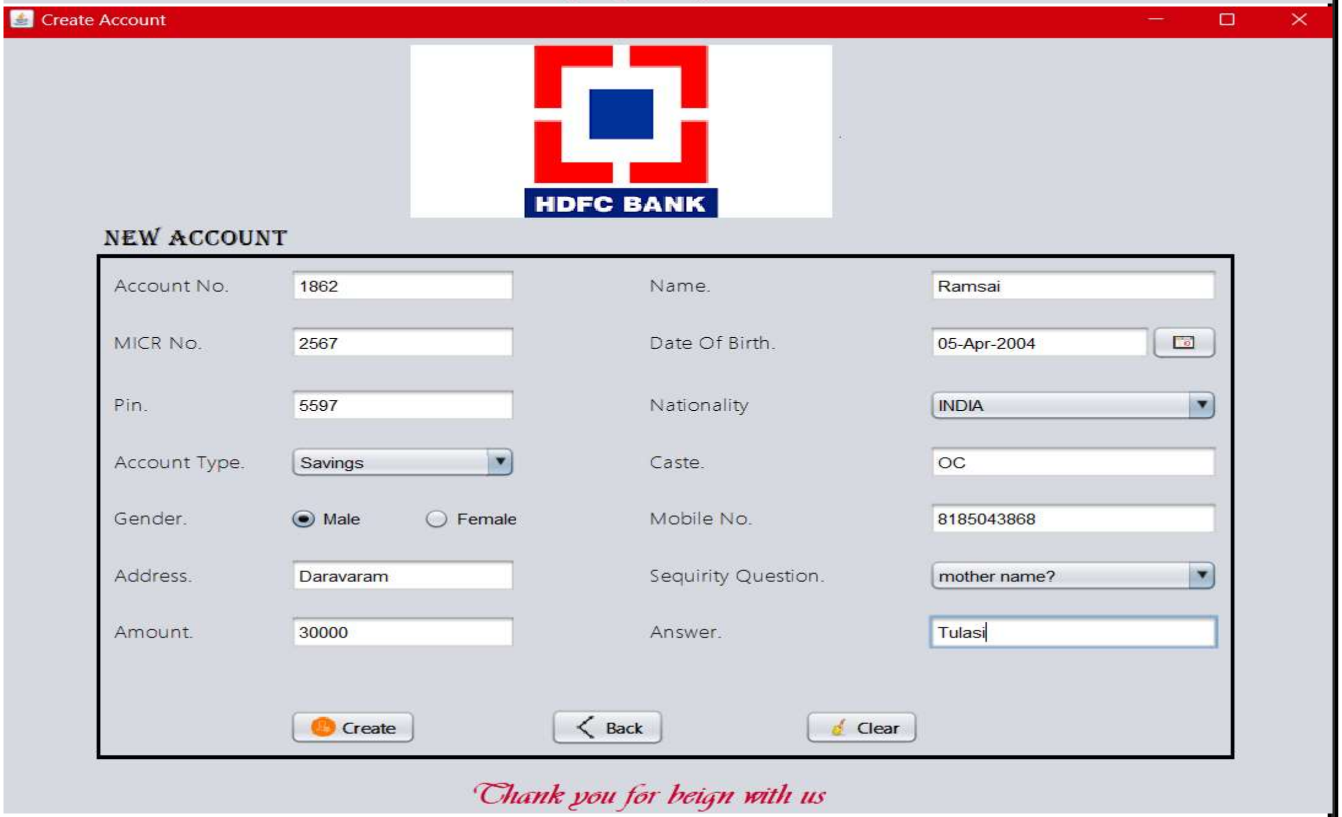
NEW ACCOUNT

Account No.	1862	Name.	Ramsai
MICR No.	2567	Date Of Birth.	
Pin.	5597		
Account Type.	Savings ▼		
Gender.	● Male ○ Female		
Address.	Daravaram		
Amount.	30000		

May ▼ 2023

	Sun	Mon	Tue	Wed	Thu	Fri	Sat
18		1	2	3	4	5	6
19	7	8	9	10	11	12	13
20	14	15	16	17	18	19	20
21	21	22	23	24	25	26	27
22	28	29	30	31			

Thank you for beign with us



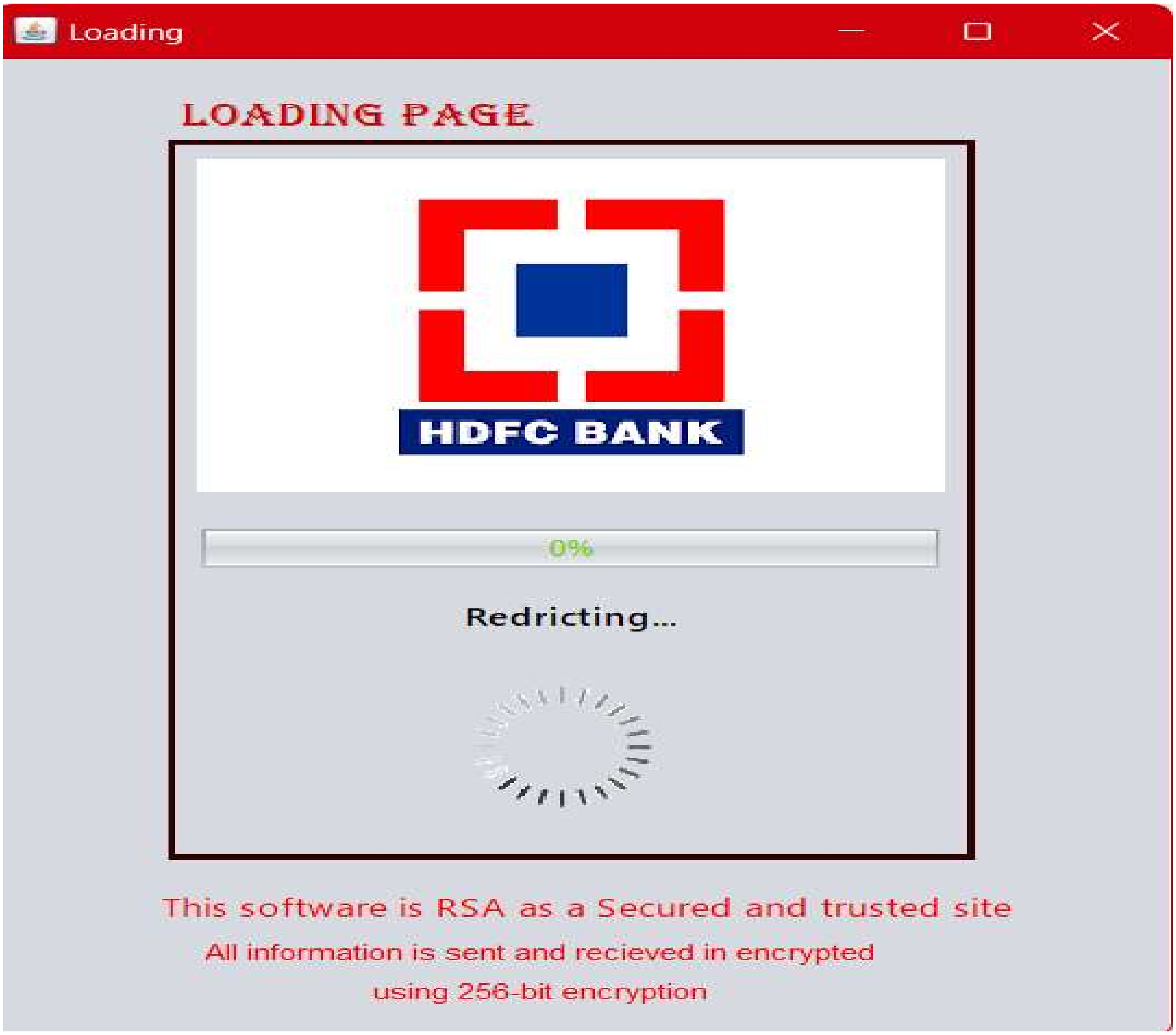
Create Account

NEW ACCOUNT

Account No.	1862	Name.	Ramsai
MICR No.	2567	Date Of Birth.	05-Apr-2004
Pin.	5597	Nationality	INDIA ▼
Account Type.	Savings ▼	Caste.	OC
Gender.	● Male ○ Female	Mobile No.	8185043868
Address.	Daravaram	Sequirity Question.	mother name? ▼
Amount.	30000	Answer.	Tulasi

Thank you for beign with us

#Loading Page:



LOADING PAGE

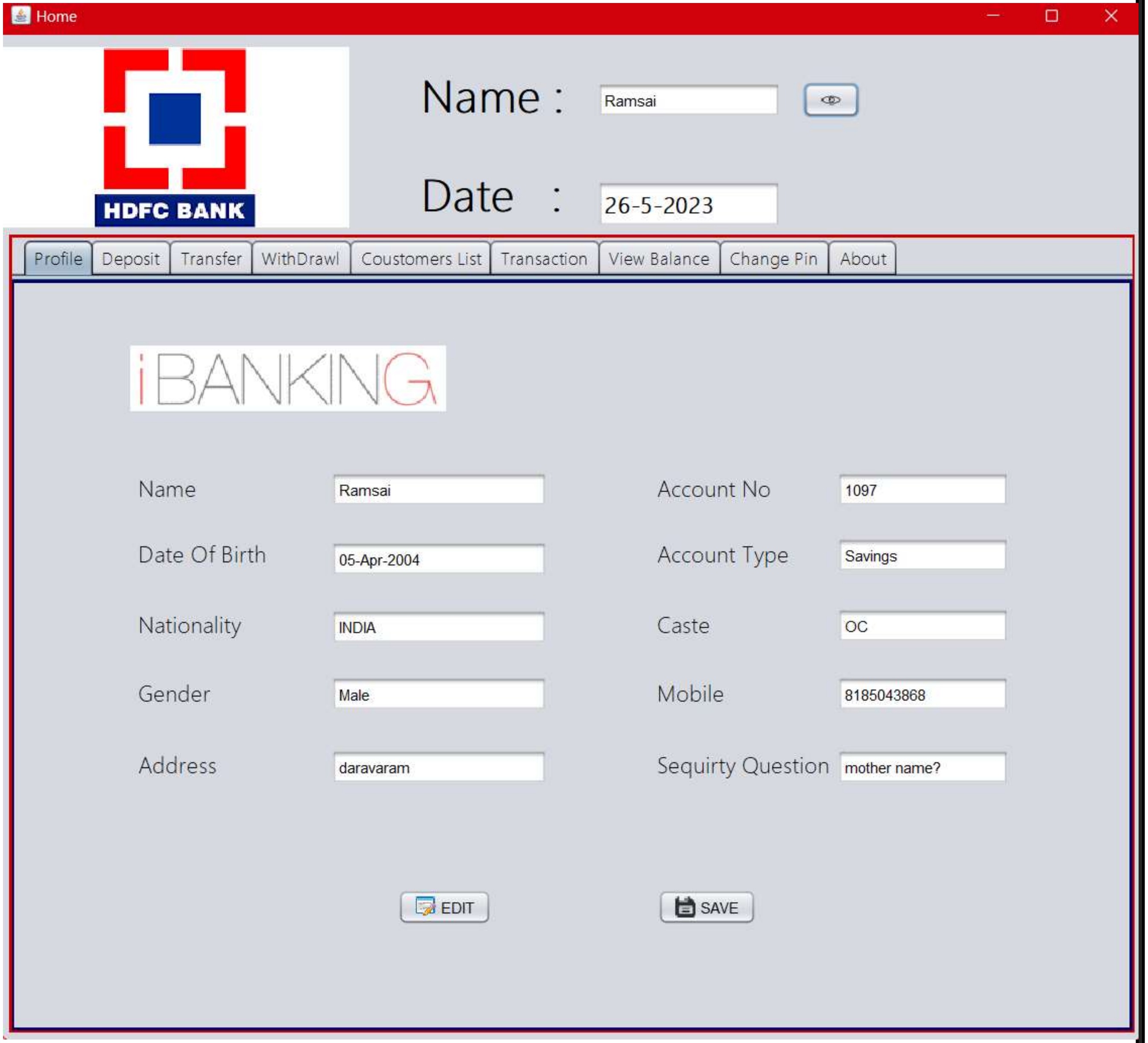
Redricting...

This software is RSA as a Secured and trusted site

All information is sent and recieved in encrypted
using 256-bit encryption

#My Page:

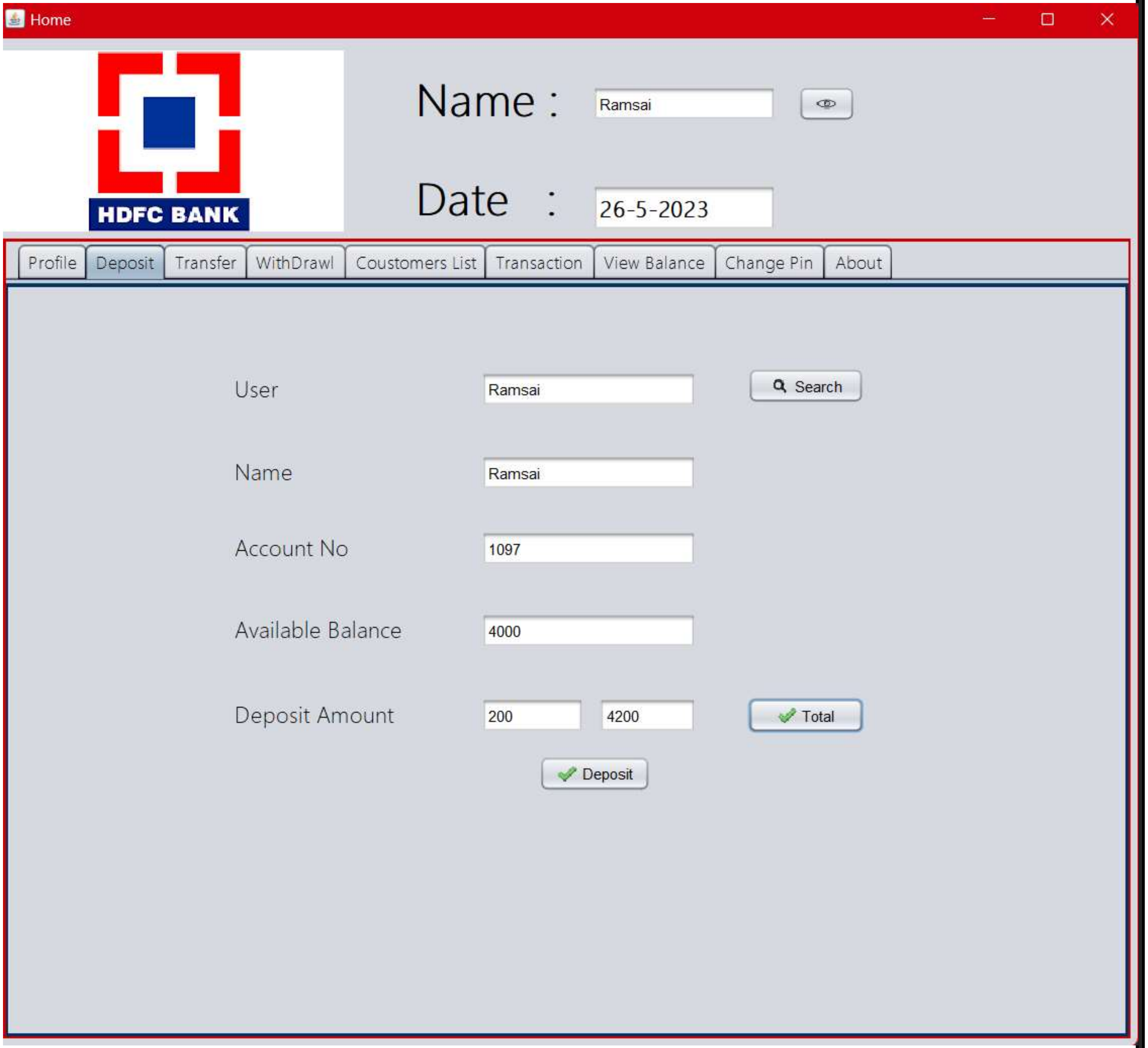
--->Profile Tab:



Name :	Ramsai
Date :	26-5-2023

Name	Ramsai	Account No	1097
Date Of Birth	05-Apr-2004	Account Type	Savings
Nationality	INDIA	Caste	OC
Gender	Male	Mobile	8185043868
Address	daravaram	Sequirty Question	mother name?

--->Deposit Tab:



Name : Ramsai

Date : 26-5-2023

User Ramsai

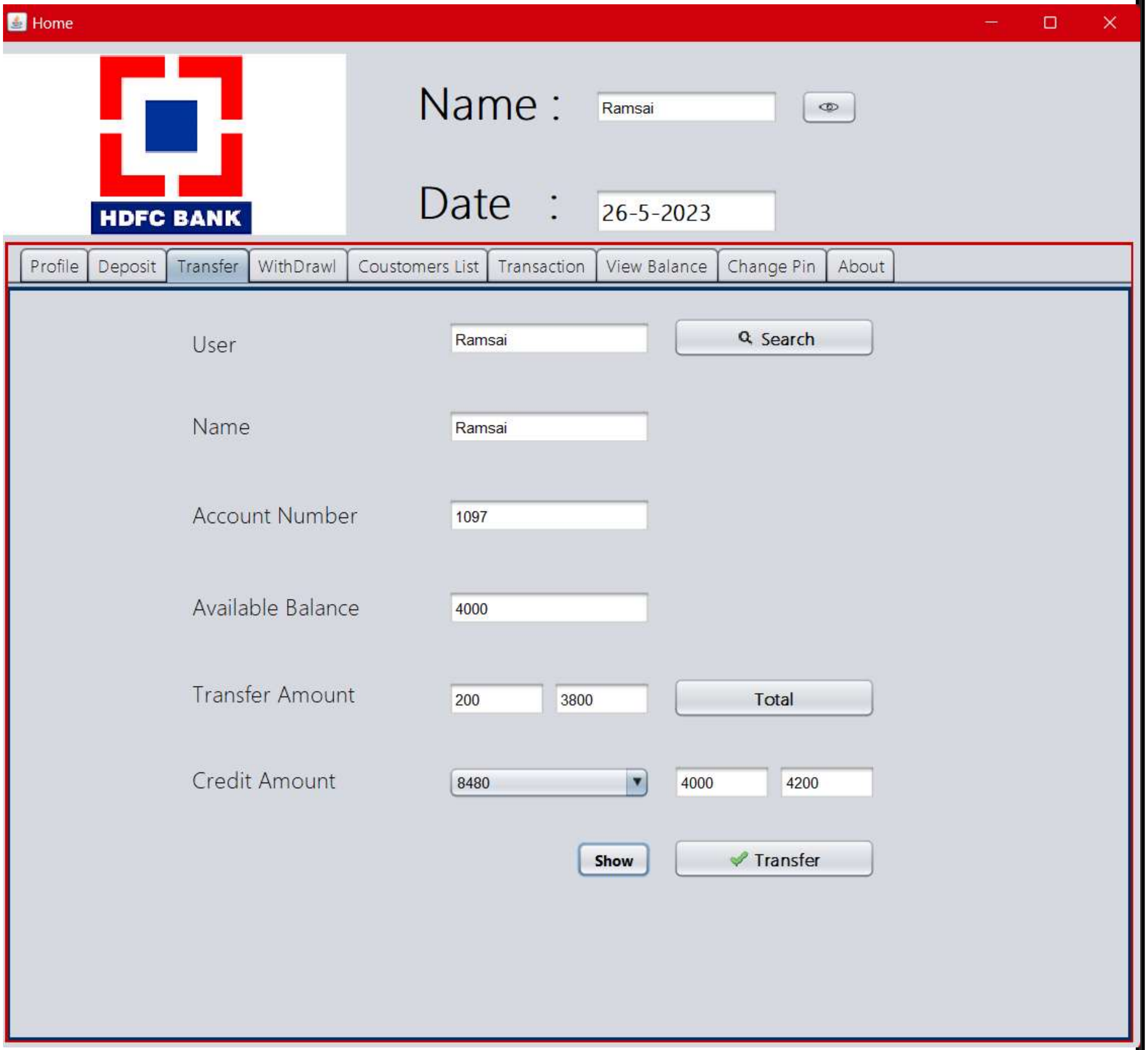
Name Ramsai

Account No 1097

Available Balance 4000

Deposit Amount 200 4200

--->Transaction Tab:

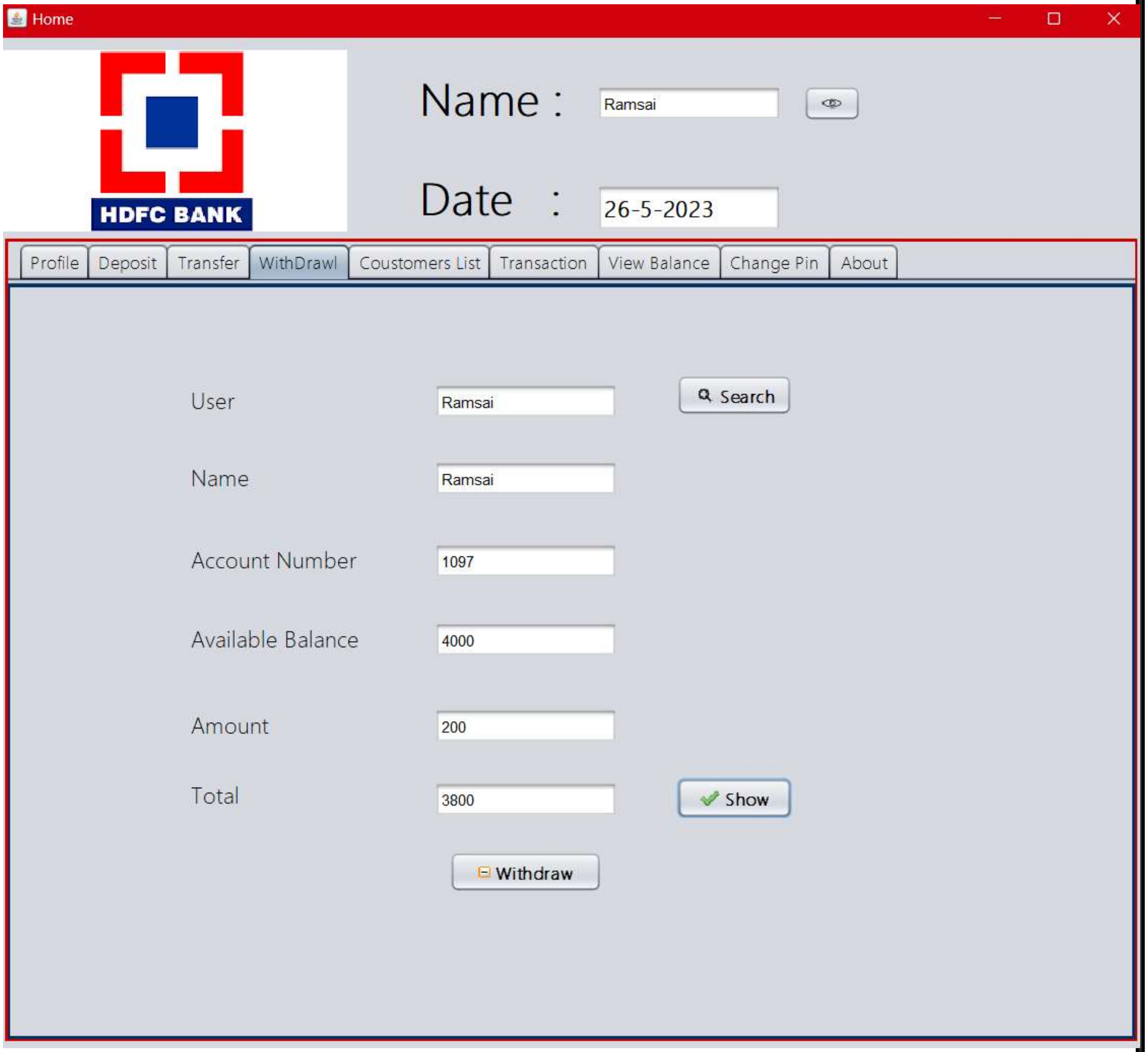


Name : Ramsai

Date : 26-5-2023

User	Ramsai
Name	Ramsai
Account Number	1097
Available Balance	4000
Transfer Amount	200 3800
Credit Amount	8480 ▾ 4000 4200

--->Withdrawl Tab:

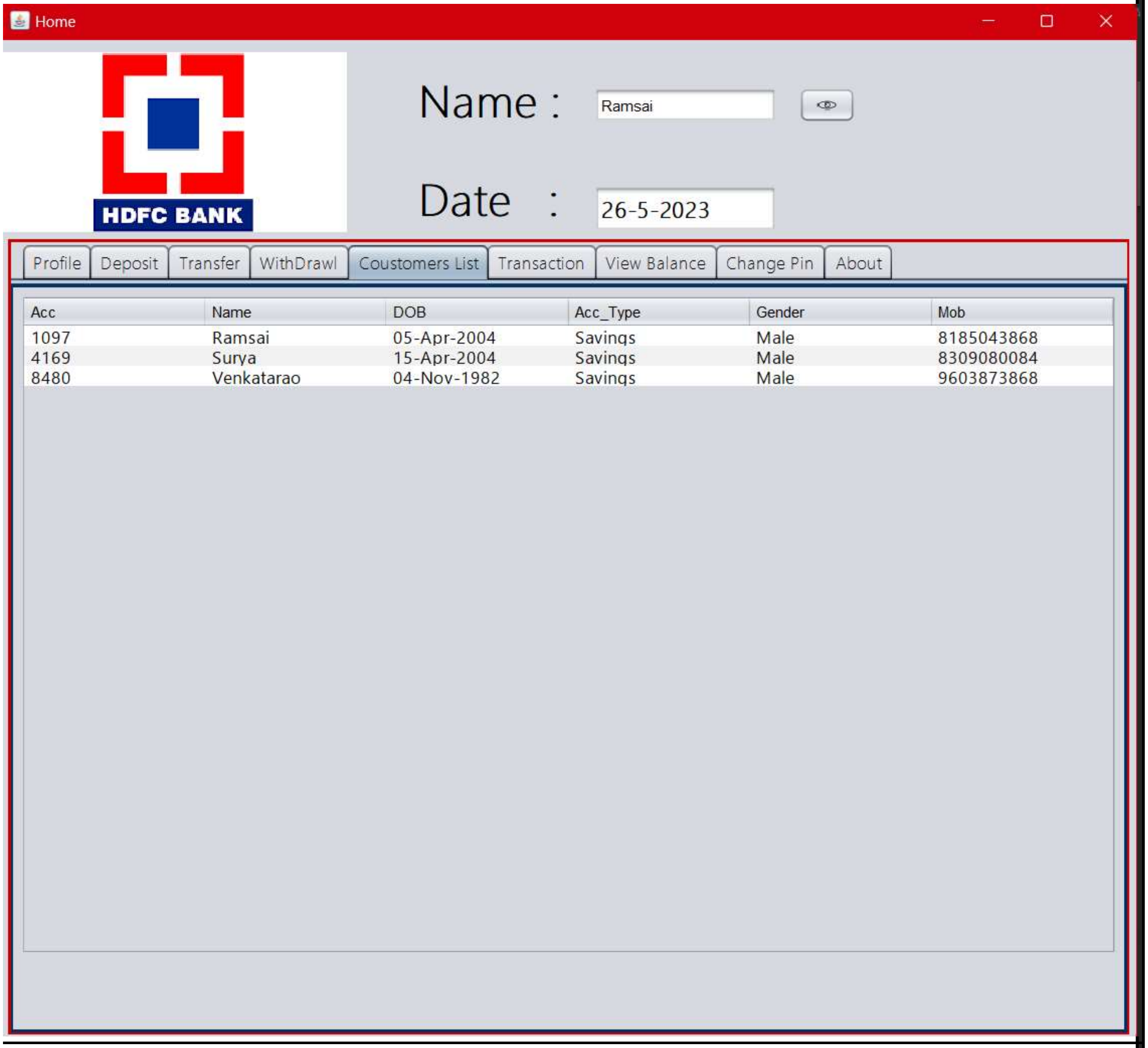


Name : Ramsai

Date : 26-5-2023

User	Ramsai
Name	Ramsai
Account Number	1097
Available Balance	4000
Amount	200
Total	3800

--->ConsumerList Tab:

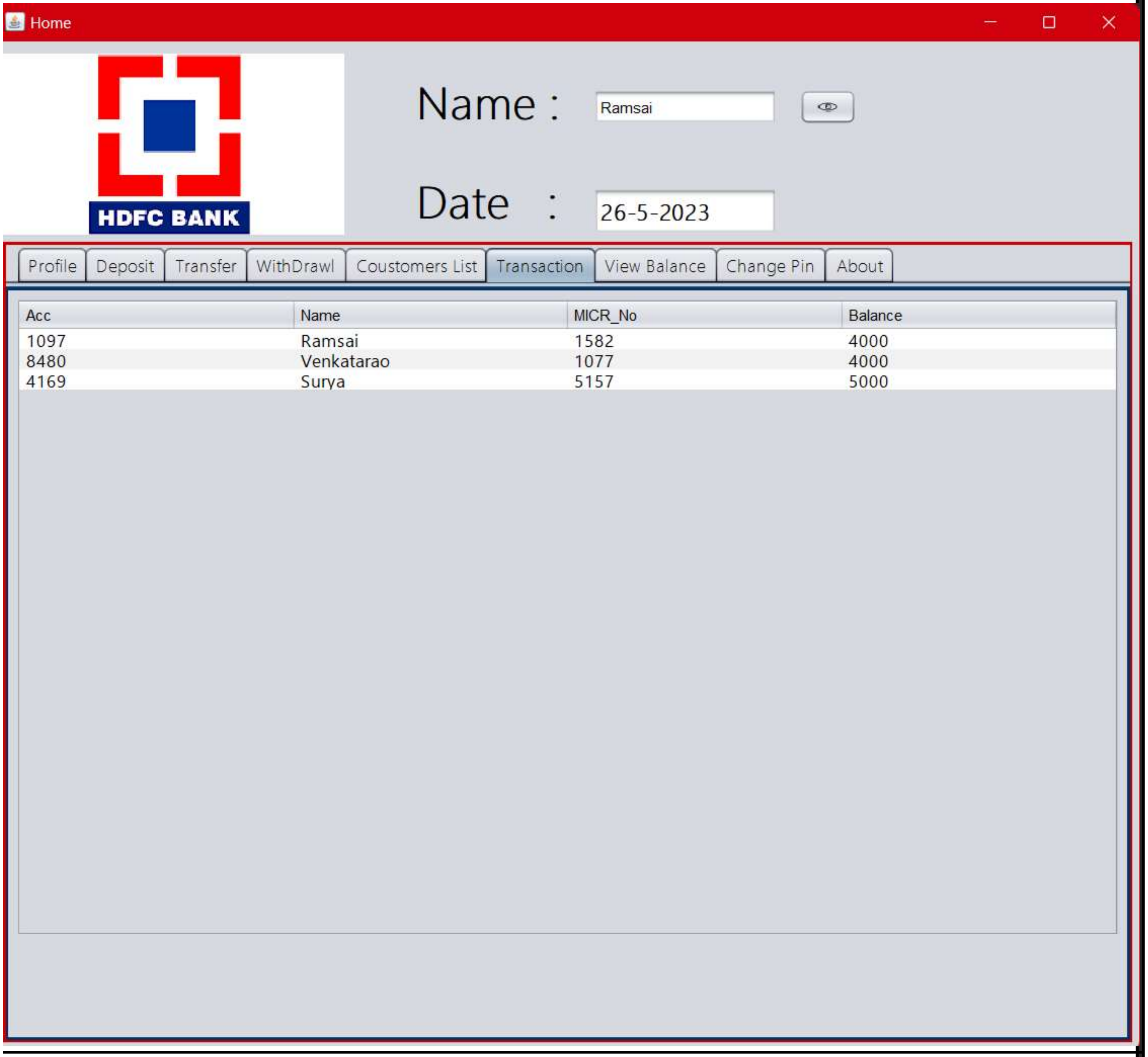


Name : Ramsai

Date : 26-5-2023

Acc	Name	DOB	Acc_Type	Gender	Mob
1097	Ramsai	05-Apr-2004	Savings	Male	8185043868
4169	Surya	15-Apr-2004	Savings	Male	8309080084
8480	Venkatarao	04-Nov-1982	Savings	Male	9603873868

--->TransactionList Tab:

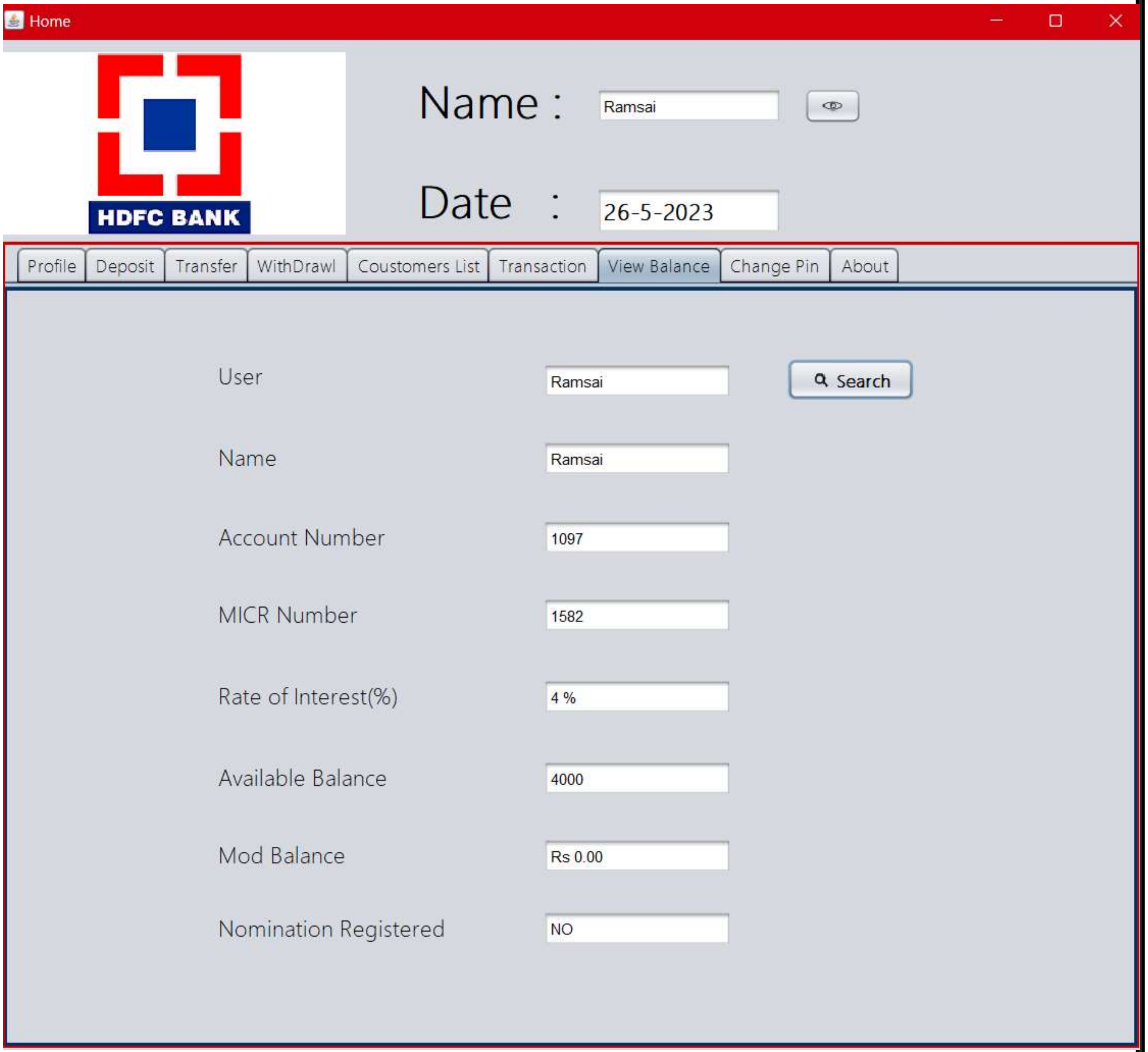


Name : Ramsai

Date : 26-5-2023

Acc	Name	MICR_No	Balance
1097	Ramsai	1582	4000
8480	Venkatarao	1077	4000
4169	Surya	5157	5000

--->View Balance Tab:



Name : Ramsai

Date : 26-5-2023

User Ramsai

Name Ramsai

Account Number 1097

MICR Number 1582

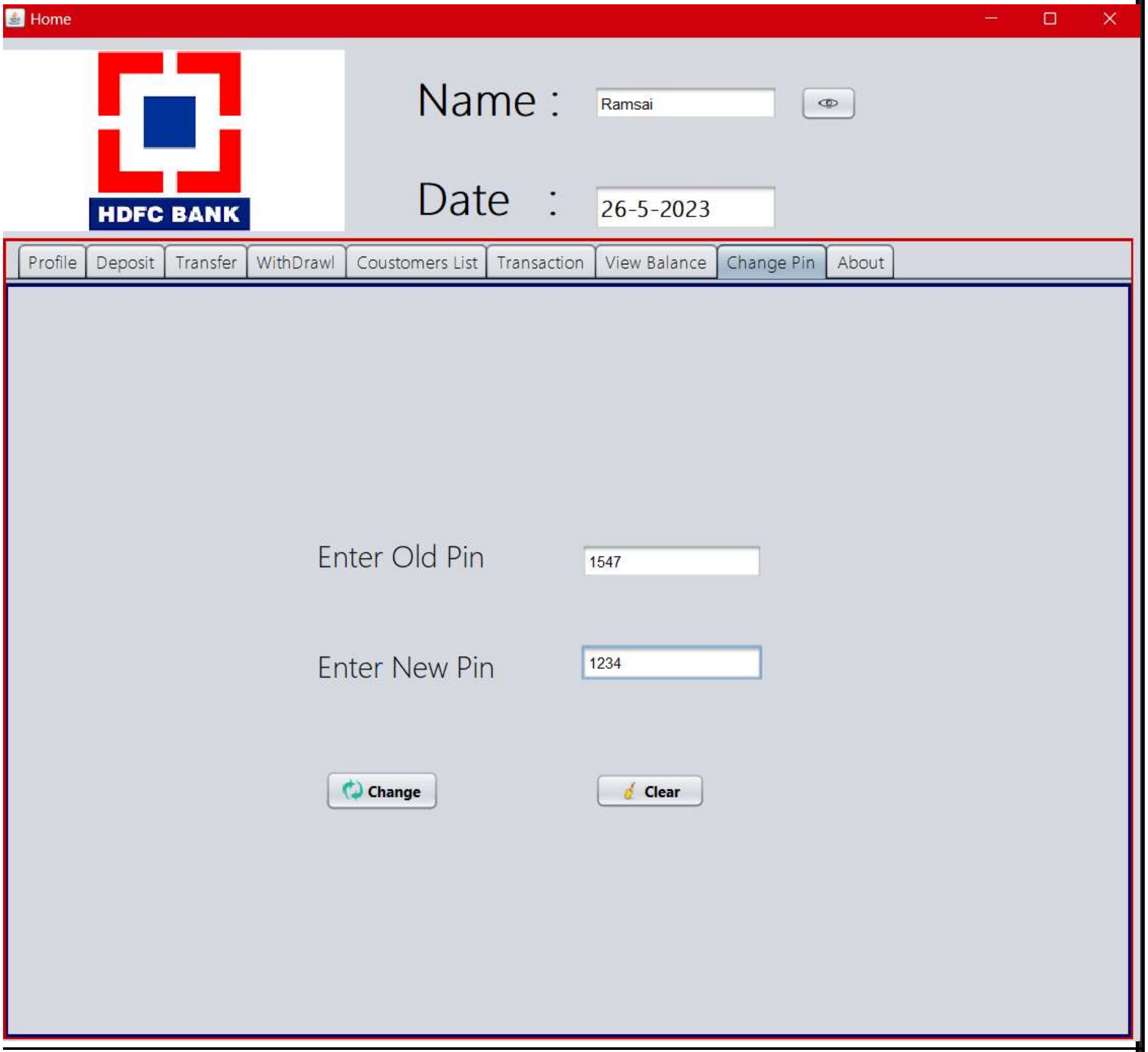
Rate of Interest(%) 4 %

Available Balance 4000

Mod Balance Rs 0.00

Nomination Registered NO

--->ChangePin Tab:



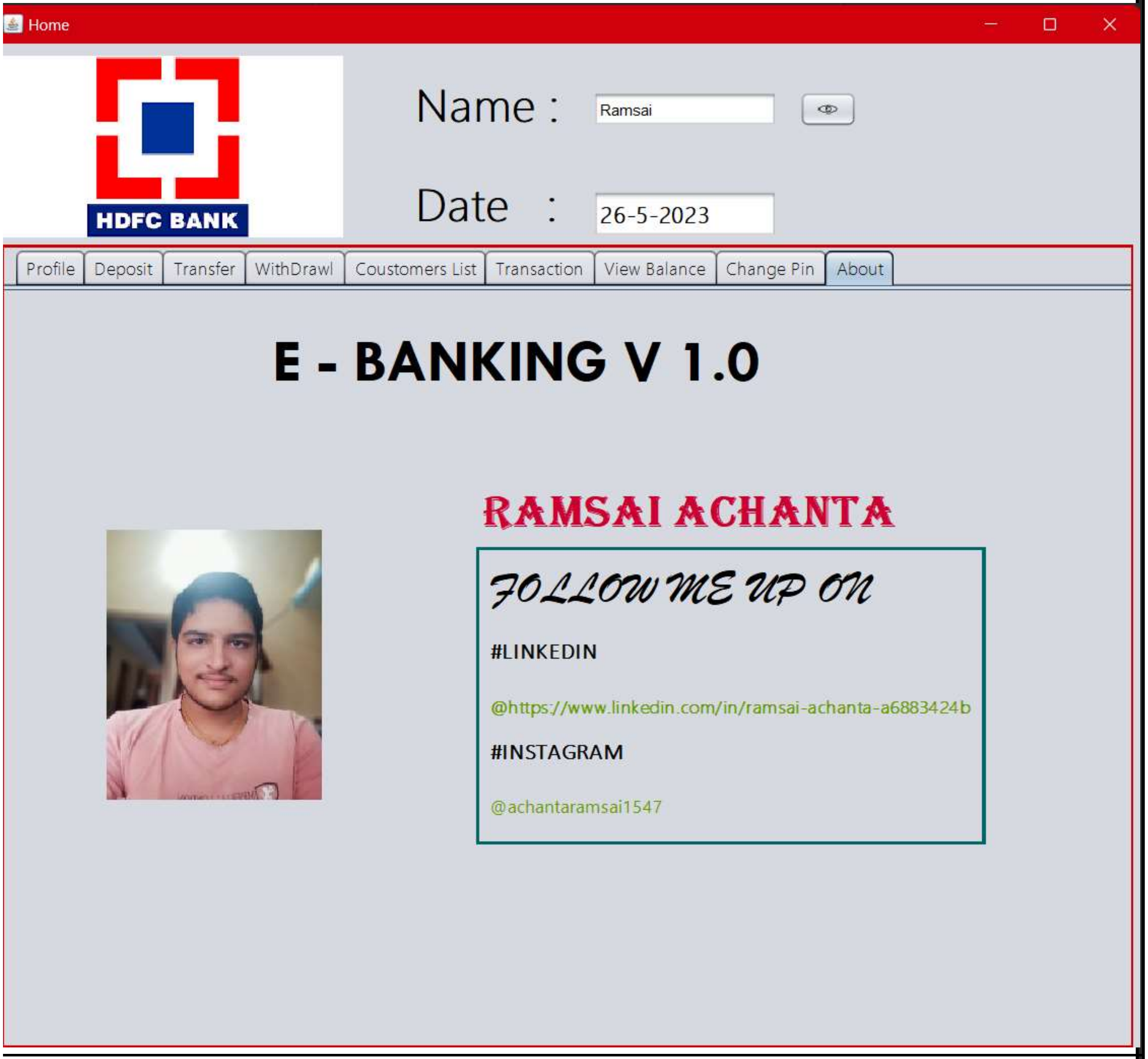
Name : Ramsai

Date : 26-5-2023

Enter Old Pin 1547

Enter New Pin 1234

--->About Tab:



Name : Ramsai

Date : 26-5-2023

E - BANKING V 1.0

RAMSAI ACHANTA

FOLLOW ME UP ON

#LINKEDIN

@https://www.linkedin.com/in/ramsai-achanta-a6883424b

#INSTAGRAM

@achantaramsai1547

Chapter: 6

ATTRACTIONS OF THE PROJECT:

1. User Interface: This Project has been Enhanced with a graphical user interface (GUI) which can provide a more user-friendly experience.

2. Account Transactions: Expanded the range of transactions, such as fund transfers between accounts or bill payments with just login of one account.

3. Transaction History: Implemented a feature to store transaction history for each account, allowing bank employees to view past transactions.

4. Account Types: Introduced different types of bank accounts, such as savings accounts, checking accounts, or investment accounts, each with its specific features and rules.

5. Error Handling: Implemented robust error handling mechanisms to handle exceptions, input validation, and error reporting to provide a more reliable system so the user wont muggle up Or get worry with any error he/she has done.

6. Reporting and Analytics: We build functionality to generate account statements, summary reports, or provide analytics on account activity and trends.

FUTURE ENHANCEMENTS OF THE PROJECT:

1. Security Enhancements: Strengthen security measures, such as implementing encryption for sensitive information, enforcing stronger password policies, or adding multi-factor authentication.

2. Multi-User Support: Enable multiple users to access the system concurrently with proper authentication and session management.

Chapter: 7
CONCLUSION

The Bank Management System provides a basic framework for managing bank accounts and performing banking operations. It demonstrates the use of classes, objects, file handling, user input/output, and basic data manipulation. The system allows users to register, login, perform transactions, and delete their account. It provides a starting point for further development and enhancement. We convey our sincere gratitude to our guide- Mr. Narala Sudhakar Reddy sir for his constant efforts to help us throughout this project.