11/25/2020

ATM INTERFACE

Ankit Choudhary	**1902029**
Mittul Daswani	**1902031**
Aarya Devnani	**1902035**

: MINI PROJECT REPORT:

Problem statement: To create an ATM interface

Theory:

A. working of the project:

we have created an ATM interface where users can create their digital accounts by either signup (new users) or login (existing users) and they can deposit or withdraw their money also users can check their balance after their latest transactions and generate bills. It is an user friendly interface where users can easily understand the process without even knowing the code.

The first page is a design where the ATM card acts as a button that triggers the second page where the signup and login options are given. If you are a new user you can signup and when you signup it takes you to the login page if you are already an user in our database you can directly login to our page.

After the login page is been filled and you have successfully been transferred to our option page there are 5 options given to our users

1. DEPOSIT
2. WITHDRAW
3. BALANCE
4. RECEIPT
5. EXIT

1. deposit: when you click on the deposit option it will take you to our new frame and you can enter the number you want to deposit. The limit of deposit is 1 lakh per turn that is each time a user can maximum deposit 1 lakh rupees to his/ her account if you try to deposit a number greater than the limit it will throw an error that “this amount cannot be deposited to you account in one sitting” and 0 rupees will be deposited. Also if you try to deposit an invalid number it will throw an error.

2.withdraw: the withdraw feature is similar to the deposit function the user can withdraw the money of limit 50000 at one go if you try to withdraw more or type an invalid number respective errors will be thrown. In addition to that user can only withdraw the money if he has an existing balance if the user tries to withdraw the money greater than the existing balance an insufficient balance error will be thrown.

3.**balance:** here the user can check the balance on his/her account after every transaction process and see if the earlier transactions were successful or not.

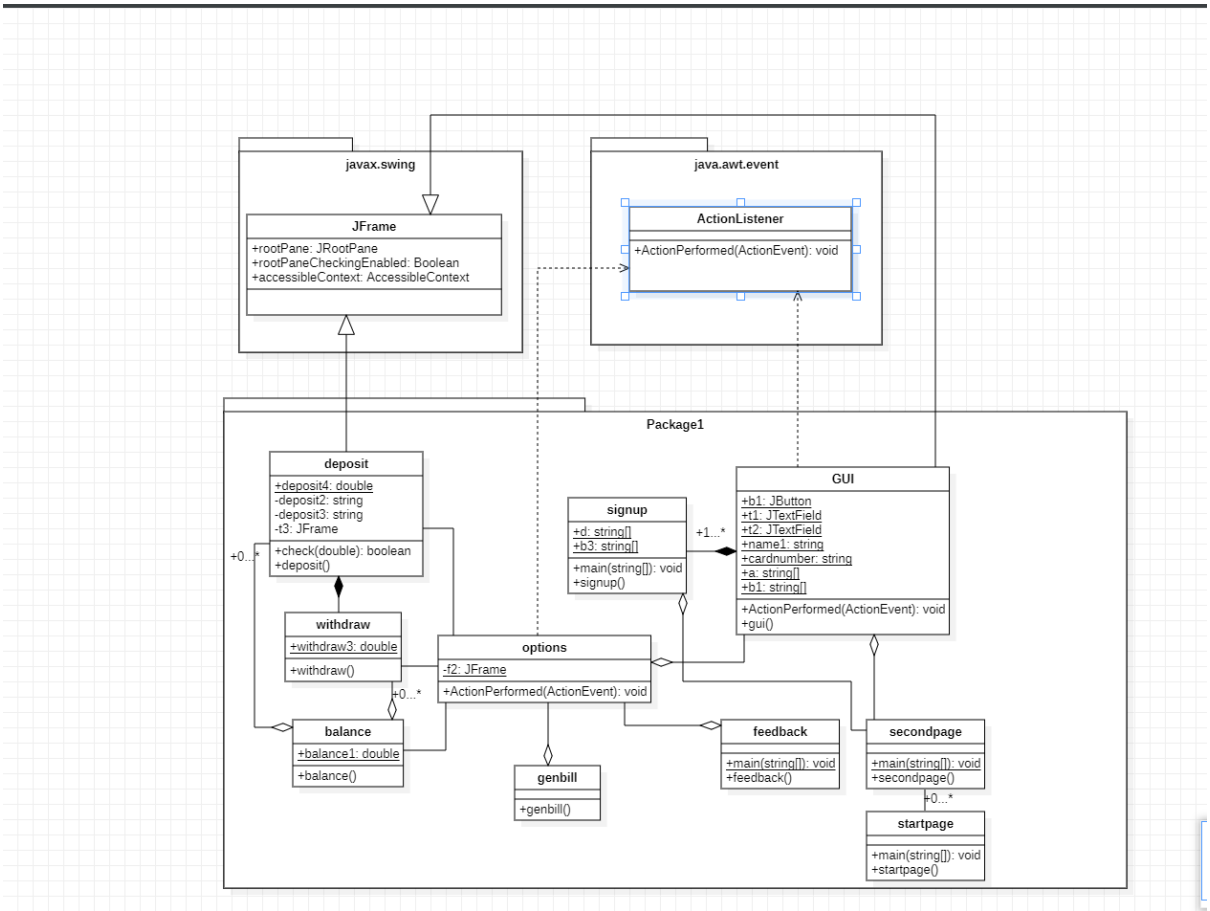
4.**receipt:** when this option is chosen the user can check every detail of the transaction and print the receipt for future reference.

5.**Exit:** when the user tries to choose this option the interface asks for a feedback form , you can either click yes or no if you click no the system directly exits , if you click yes it asks about the experience with the interface and you can submit your suggestions to us.

B. features of java used:

- Swing
- awt
- Array of objects
- Strings
- Packages
- Inbuilt exceptions handling
- Methods
- Objects
- Classes
- Access specifiers
- Conditional branching statements
- Constructor
- Inheritance for swings
- Interface

UML diagram:



Code:

Package programs

Startpage:

```
package package1;


import javax.swing.*;


public class startpage {
    public startpage() {
        JFrame f = new JFrame("START PAGE");
        JLabel l = new JLabel();
        JButton b = new JButton();
        b.setIcon(new ImageIcon("C:\\Users\\Ankit\\Desktop\\oopatm\\package1\\card.jpeg"));
        l.setIcon(new
ImageIcon("C:\\Users\\Ankit\\Desktop\\oopatm\\package1\\starticon.jpeg"));
        l.setBounds(0, 0, 554, 554);
        b.setBounds(60, 500, 320, 520);
        b.addActionListener(e1 -> {
            f.dispose();
            new secondpage();
        });
        f.add(l);
        f.add(b);
        f.setSize(554, 1100);
        f.setLayout(null);


        f.setVisible(true);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);


    }


    public static void main(String[] args) {
```

```
        new startpage();



    }
}
```

Second page:

```
package package1;

import javax.swing.*;

public class secondpage {
    secondpage(){
        JFrame f= new JFrame("login signup");
        JButton b1= new JButton("login");
        JButton b2=new JButton("signup");
        b1.setBounds(30,100,80,30);
        b2.setBounds(160,100,80,30);

        b1.addActionListener(e->{

            f.dispose();
            new GUI();
        });
        b2.addActionListener(e->{
            f.dispose();
            new signup();
        });
        f.add(b1);
        f.add(b2);
```

```
        f.setSize(310, 350);
        f.setLayout(null);
        f.setVisible(true);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public static void main(String[] args) {
        new secondpage();
    }
}
```

Signup page:

```
package package1;
import javax.swing.*;
import java.util.*;
import java.util.Arrays;

public class signup {
    public static String d[];
    public static String b3[];
    {
        Scanner sc =new Scanner(System.in);
        String[] c=GUI.a;
        d= Arrays.copyOf(c,c.length+1);
        String[] b2=GUI.b1;
        b3=Arrays.copyOf(b2,b2.length+1);
        JFrame f=new JFrame("signup");
        JLabel l1= new JLabel("Enter Name");
        l1.setBounds(20,50,150,30);
        JTextField t1=new JTextField();
```

```
    t1.setBounds(150,50,150,30);
    JLabel l2= new JLabel("Enter Cardnumber");
    l2.setBounds(20,100,150,30);
    JTextField t2=new JTextField();
    t2.setBounds(150,100,150,30);
    JButton b= new JButton("signup");

    b.setBounds(150,150,80,30);
    b.addActionListener(e-> {
       if(t1.getText().equals("")||t2.getText().equals(""))
       {
          JOptionPane.showMessageDialog(null, "enter credentials",
                "ERROR", JOptionPane.ERROR_MESSAGE);
       }
       else {
       d[3] = t1.getText();
       b3[3] = t2.getText();
       new GUI();
       f.dispose();
    }

    });
    f.add(l1);f.add(l2);f.add(t1);f.add(t2);f.add(b);
    f.setSize(400, 400);
    f.setLayout(null);
    f.setVisible(true);
    f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

}
```

```
    public static void main(String[] args) {
        new signup();
    }
}
```

GUI page:

```
package package1;

import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class GUI extends JFrame implements ActionListener {
    public JButton b;
    JFrame f;
    public static JTextField t1;
    public static JTextField t2;
    public static String name1;
    public static String cardnumber;
    public static String[] a={"Ankit Choudhary", "Mittul Daswani", "Aarya Devanai"};
   public static String[] b1 = {"4375 1111 2222 1234", "1234 5678 9012 3456", "4322 1232
0000 0001"};
    public GUI() {
        f = new JFrame("Login");
        JLabel l1, l2;
        l1 = new JLabel("Enter Name");
        l1.setBounds(50, 50, 100, 30);
        t1 = new JTextField();
        t1.setBounds(180, 50, 150, 30);
        l2 = new JLabel("Enter Cardnumber");
        l2.setBounds(50, 100, 150, 70);
```

```
        t2 = new JTextField();
        t2.setBounds(180, 120, 200, 30);
        b = new JButton("Submit");
        b.setBounds(135, 190, 110, 30);
        b.addActionListener(this);
        f.add(b);
        f.add(l1);
        f.add(l2);
        f.add(t1);
        f.add(t2);
        f.setSize(400, 400);
        f.setLayout(null);

        f.setVisible(true);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public void actionPerformed(ActionEvent e) {

        name1 = t1.getText();



        cardnumber = t2.getText();
        String[] k ;
        String[] l;
        k=signup.d;
        l=signup.b3;
        if (((name1.equals(a[0])) && (cardnumber.equals(b1[0]))) || ((name1.equals(a[1])) &&
(cardnumber.equals(b1[1]))) || ((name1.equals(a[2])) &&
(cardnumber.equals(b1[2])))||((name1.equals(k[3])) && (cardnumber.equals(l[3])))) {
            try {
```

```
            Thread.sleep(1000);
        } catch (InterruptedException interruptedException) {
            interruptedException.printStackTrace();
        }
        new options();
        f.dispose();
    } else {
        JOptionPane.showMessageDialog(null, "not registered in our database",
                "ERROR", JOptionPane.ERROR_MESSAGE);

    }

  }

}
```

Options:

```
package package1;

import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class options implements ActionListener {
   JFrame f2;

   options() {

      f2 = new JFrame("the options we provide");
      JButton b1 = new JButton("Deposit");
      JButton b2 = new JButton("Withdraw");
```

```
        JButton b3 = new JButton("Balance");
        JButton b4 = new JButton("Receipt");
        JButton b5 = new JButton("EXIT");
        JLabel l1 = new JLabel();
        JLabel l6 = new JLabel();
        JLabel l7 = new JLabel();
        JLabel l8 = new JLabel();
        JLabel l9 = new JLabel();

        l1.setBounds(300, 220, 180, 80);
        l6.setBounds(300, 70, 180, 80);
        l6.setIcon(new
ImageIcon("C:\\Users\\Ankit\\Desktop\\oopatm\\package1\\depositicon.jpeg"));
        l7.setBounds(300, 370, 180, 80);
        l7.setIcon(new
ImageIcon("C:\\Users\\Ankit\\Desktop\\oopatm\\package1\\balanceicon.jpeg"));
        l8.setBounds(300, 520, 180, 80);
        l8.setIcon(new
ImageIcon("C:\\Users\\Ankit\\Desktop\\oopatm\\package1\\receipt.jpeg"));
        l9.setBounds(300, 670, 180, 80);
        l9.setIcon(new
ImageIcon("C:\\Users\\Ankit\\Desktop\\oopatm\\package1\\exiticon.jpeg"));
        ImageIcon with;
        with = new
ImageIcon("C:\\Users\\Ankit\\Desktop\\oopatm\\package1\\withdrawicon.jpeg");
        l1.setIcon(with);

        b1.setBounds(100, 70, 160, 80);
        b1.addActionListener(e14 -> {
            new deposit();
            f2.dispose();
        });
```

```
b2.setBounds(100, 220, 160, 80);
b2.addActionListener(e13 -> {

   new withdraw();
   f2.dispose();

});
b3.setBounds(100, 370, 160, 80);
b3.addActionListener(e12 -> {
   new balance();
});
b4.setBounds(100, 520, 160, 80);
b4.addActionListener(e17 -> {
   new  genbill();
   f2.dispose();



});
b5.setBounds(100, 670, 160, 80);
b5.addActionListener(e1 -> {
   int result = JOptionPane.showConfirmDialog(null,"would you like to fill a feedback
form", "Swing Tester",
         JOptionPane.YES_NO_OPTION,
         JOptionPane.QUESTION_MESSAGE);
   if(result == JOptionPane.YES_OPTION){
      f2.dispose();
     new feedback();
   }
   else {
      System.exit(0);
   }
```

```
        });
        f2.add(b1);
        f2.add(b2);
        f2.add(b3);
        f2.add(b4);
        f2.add(b5);
        f2.add(l1);
        f2.add(l6);
        f2.add(l7);
        f2.add(l8);
        f2.add(l9);


        f2.setSize(550, 800);
        f2.setLayout(null);

        f2.setVisible(true);

        f2.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    @Override
    public void actionPerformed(ActionEvent e) {



    }
}
```

Deposit:

```
package package1;
```

```
import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class deposit extends JFrame {
    public static double deposit4 = 0;
    private String deposit2 = "";
     private double deposit3 = 0;
   private  JFrame f3;

    public deposit() {

        f3 = new JFrame("DEPOSIT");
        JButton b = new JButton("DEPOSIT");
        JButton b2 = new JButton("Back");
        JLabel l4 = new JLabel("Enter the amount you want to deposit");
        l4.setBounds(40, 50, 300, 60);
        JTextField t3 = new JTextField();
        t3.setBounds(100, 100, 100, 30);
        b.setBounds(60, 150, 90, 30);
        b2.setBounds(180, 150, 80, 30);
        b.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                f3.dispose();
                deposit2 = t3.getText();
                deposit3 = Double.parseDouble(deposit2);
                if (deposit3 > 100000) {

                    JOptionPane.showMessageDialog(null, "this amount cannot be deposited in one
sitting",
                            "ERROR", JOptionPane.ERROR_MESSAGE);
```

```
            } else if (deposit3 < 0) {
                JOptionPane.showMessageDialog(null, "invalid number",
                        "ERROR", JOptionPane.ERROR_MESSAGE);
            } else {
                deposit4 = deposit3;
                balance.balance1 = balance.balance1 + deposit4;
                JOptionPane.showMessageDialog(null, "rupees " + deposit4 + " has been
deposited");
                System.out.println("rupees:" + deposit4 + " has been deposited");
                f3.dispose();



            }
            new options();

        }
    });

    b2.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            f3.dispose();
            new options();
        }
    });



    f3.add(b);
    f3.add(b2);
    f3.add(l4);
    f3.add(t3);
```

```
        f3.setSize(400, 400);
        f3.setLayout(null);
        f3.setVisible(true);
        f3.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public static boolean check(double abc) {


        if (abc > balance.balance1) {
            System.out.println(balance.balance1);
            return true;

        } else {
            return false;
        }

    }
}
```

Withdraw:

```
package package1;



import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class withdraw {

    public static double withdraw3 = 0;
```

```
public withdraw() {
    JFrame f3 = new JFrame("withdraw");
    JButton b = new JButton("Withdraw");
    JButton b2 = new JButton("Back");
    JLabel l4 = new JLabel("Enter the amount you want to withdraw");
    l4.setBounds(40, 50, 300, 60);
    JTextField t3 = new JTextField();
    t3.setBounds(100, 100, 100, 30);
    b.setBounds(80, 150, 90, 30);
    b2.setBounds(190, 150, 80, 30);
    b.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            f3.dispose();



            String withdraw1 = t3.getText();
            double withdraw2 = Double.parseDouble(withdraw1);


            if (withdraw2 > 50000) {
                JOptionPane.showMessageDialog(null, "this amount cannot be withdrawn in one
sitting",
                        "ERROR", JOptionPane.ERROR_MESSAGE);
                System.out.println("rupees" + withdraw2 + "cannot be withdrawn at one go
please try again later");
            } else if ((deposit.check(withdraw2) == true)) {


                JOptionPane.showMessageDialog(null, "insufficent balance",
                        "ERROR", JOptionPane.ERROR_MESSAGE);


                System.out.println("error");
```

```
            }
            else if((withdraw2 < 0)){
               JOptionPane.showMessageDialog(null, "error",
                     "ERROR", JOptionPane.ERROR_MESSAGE);
            }
            else {
               double withdraw3 = withdraw2;
               balance.balance1 = balance.balance1 - withdraw3;
               JOptionPane.showMessageDialog(null, "rupees " + withdraw3 + " has been
withdrawn");
               System.out.println("rupees" + withdraw3 + "is withdrawn");
            }

            new options();
         }
      });
      b2.addActionListener(new ActionListener() {
         @Override
         public void actionPerformed(ActionEvent e) {
            f3.dispose();
            new options();
         }
      });

      f3.add(b);
      f3.add(b2);
      f3.add(l4);
      f3.add(t3);
      f3.setSize(400, 400);
      f3.setLayout(null);
      f3.setVisible(true);
```

```
        f3.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}
```

Balance:

```
package package1;

import javax.swing.JOptionPane;

public class balance {
    static double balance1 = 0;

    balance() {
        JOptionPane.showMessageDialog(null, "balance is rupees: " + balance1);
        System.out.println("balance amount is rupees" + balance1);
    }
}
```

Genbill:

```
package package1;

import javax.swing.*;
import java.awt.print.PrinterException;
import java.awt.print.PrinterJob;
import java.util.Calendar;


public class genbill {

    genbill() {
```

```
JFrame f5 = new JFrame("BILL");
JLabel l5 = new JLabel("NAME:");
l5.setBounds(20, 40, 100, 20);

JLabel l6 = new JLabel();
l6.setText(GUI.name1);
l6.setBounds(130, 40, 100, 20);
JLabel l7 = new JLabel("CardNumber:");
l7.setBounds(20, 70, 120, 20);
JLabel l8 = new JLabel();
l8.setText(GUI.cardnumber);
l8.setBounds(130, 70, 150, 20);
JLabel l9 = new JLabel();
l9.setText("Account Balance:");
l9.setBounds(20, 100, 150, 20);
JLabel l10 = new JLabel();
l10.setText(String.valueOf(balance.balance1));
l10.setBounds(130, 100, 150, 20);
Calendar calendar = Calendar.getInstance();
JLabel l11 = new JLabel();
l11.setText("Time of bill:");
l11.setBounds(20, 130, 150, 20);
JLabel l12 = new JLabel();
l12.setText(String.valueOf(calendar.getTime()));
l12.setBounds(130, 130, 200, 20);
System.out.println("time of bill= " + calendar.getTime());
JButton b = new JButton("print");
b.setBounds(200, 160, 90, 30);
b.addActionListener(e1 -> {
    PrinterJob printJob = PrinterJob.getPrinterJob();
```

```
    if (printJob.printDialog()) {
        try {
            printJob.print();
        } catch (PrinterException prt) {
            prt.printStackTrace();
        }
    }
});
JButton b1 = new JButton("back");
b1.setBounds(110, 160, 80, 30);
b1.addActionListener(e1 -> {
    f5.dispose();
    new options();
});
f5.add(l5);
f5.add(l6);
f5.add(l7);
f5.add(l8);
f5.add(l9);
f5.add(l10);
f5.add(l11);
f5.add(l12);
f5.add(b);
f5.add(b1);
f5.setSize(400, 400);
f5.setLayout(null);
f5.setVisible(true);
f5.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
    }
}
```

Feedback:

```
package package1;

import javax.swing.*;



public class feedback {
    feedback()  {
        JFrame f=new JFrame("feedback form");
        JLabel l=new JLabel("Did you like your experience with us.");
        JLabel l2 = new JLabel();
        JLabel l3 = new JLabel();
        JTextArea area= new JTextArea();
        JCheckBox checkbox1 = new JCheckBox("yes");
        JCheckBox checkbox2 = new JCheckBox("no");
        l.setBounds(50,40,300,30);
        checkbox1.setBounds(150,80,50,50);
        checkbox2.setBounds(150,140,50,50);
        JButton b=new JButton("submit");
        JButton b2=new JButton("submit");
        b.setBounds(120,200,100,30);
        b.addActionListener(e->{
            if(checkbox2.isSelected())
            {
                checkbox1.setSelected(false);
                l2.setText("Would you like to give some suggestions");
                l2.setBounds(30,250,300,30);
```

```
        area.setBounds(20,300,250,360);
        b2.setBounds(120,700,100,30);
        b2.addActionListener(e13->{
            if(area.getText().equals(""))
            { JOptionPane.showMessageDialog(null,"we are sorry for your experience");}
            else{
                JOptionPane.showMessageDialog(null,"your suggestions are noted");
            }
            System.exit(0);
        });



    }
    if(checkbox1.isSelected())
    {
        checkbox2.setSelected(false);
        JOptionPane.showMessageDialog(null,"Thank you");
        System.exit(0);


    }
});
f.add(l);
f.add(l2);
f.add(l3);
f.add(area);
f.add(checkbox1);
f.add(checkbox2);
f.add(b);
f.add(b2);
f.setLayout(null);
```

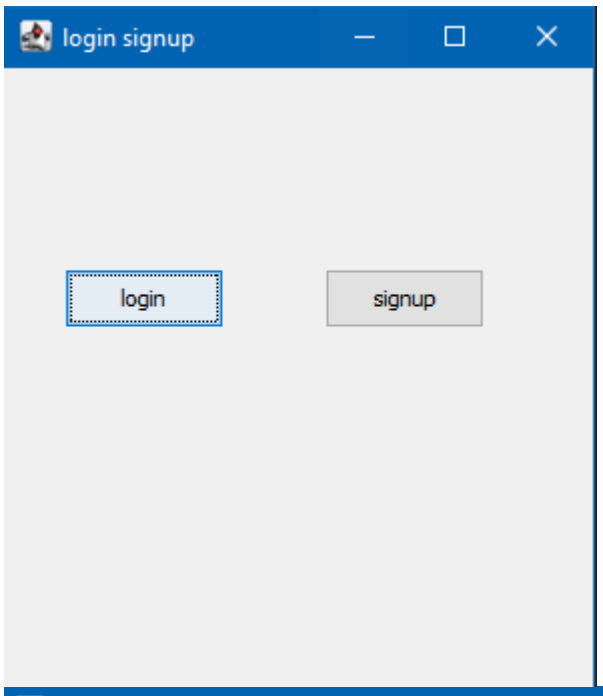
```
        f.setVisible(true);
        f.setSize(400,800);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public static void main(String[] args) throws InterruptedException {
        new feedback();
    }
}
```

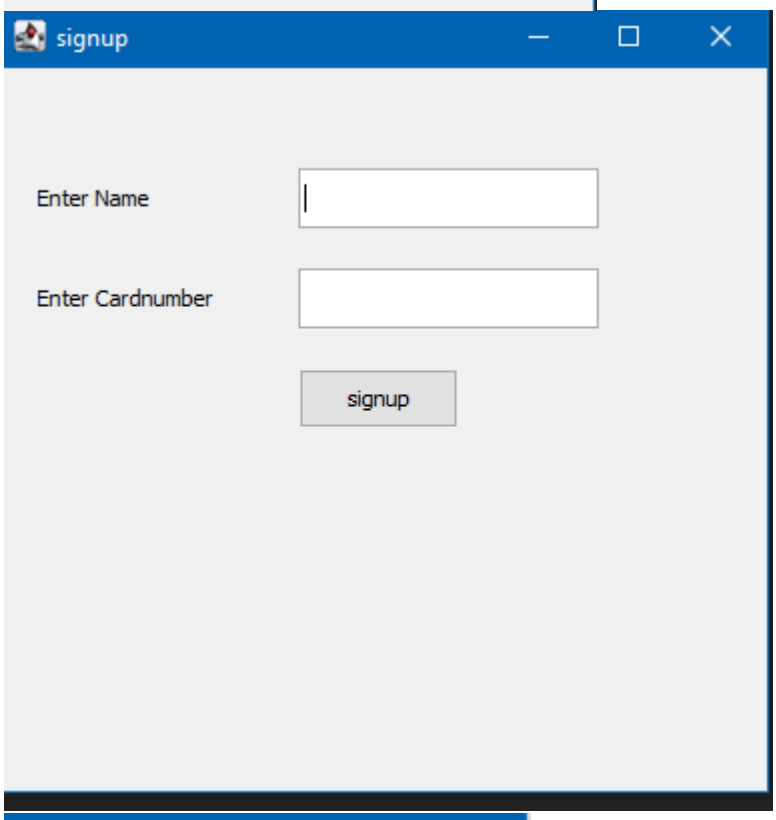
Main program:

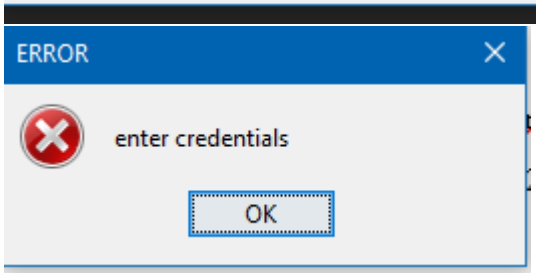
```
import package1.*;

import  javax.swing.*;

public class mainclass {
    public static void main(String[] args) {
        try {
            UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
        }
        catch (Exception e){
            e.printStackTrace();
        }
        new startpage();
    }
}
//    String[] a = {"Ankit Choudhary","Mittul Daswani","Aarya Devanai"};
//    String[] b={"4375 1111 2222 1234","1234 5678 9012 3456","4322 1232 0000 0001"};
```

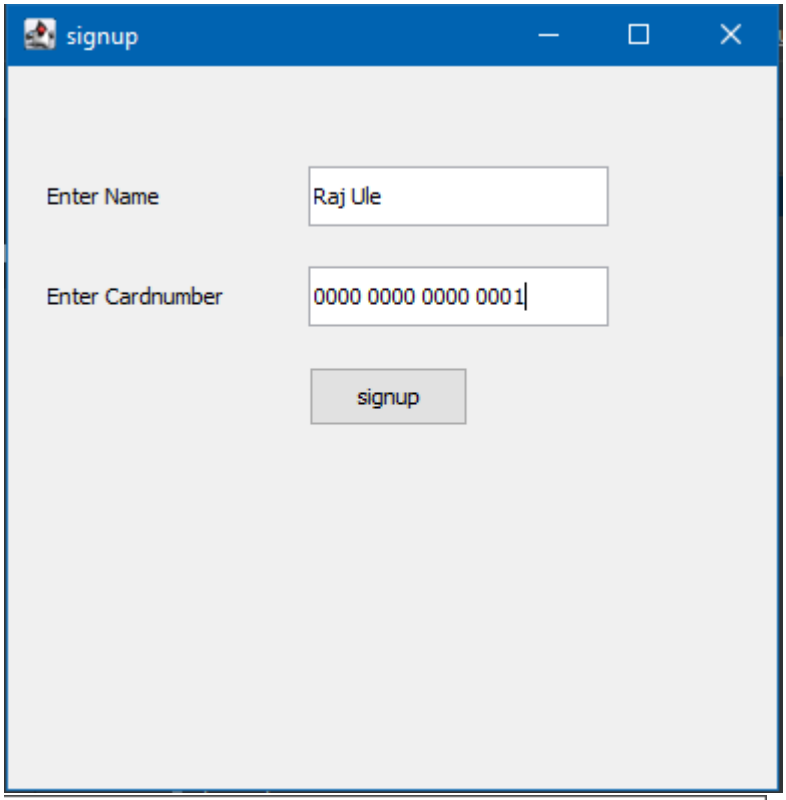
Output:

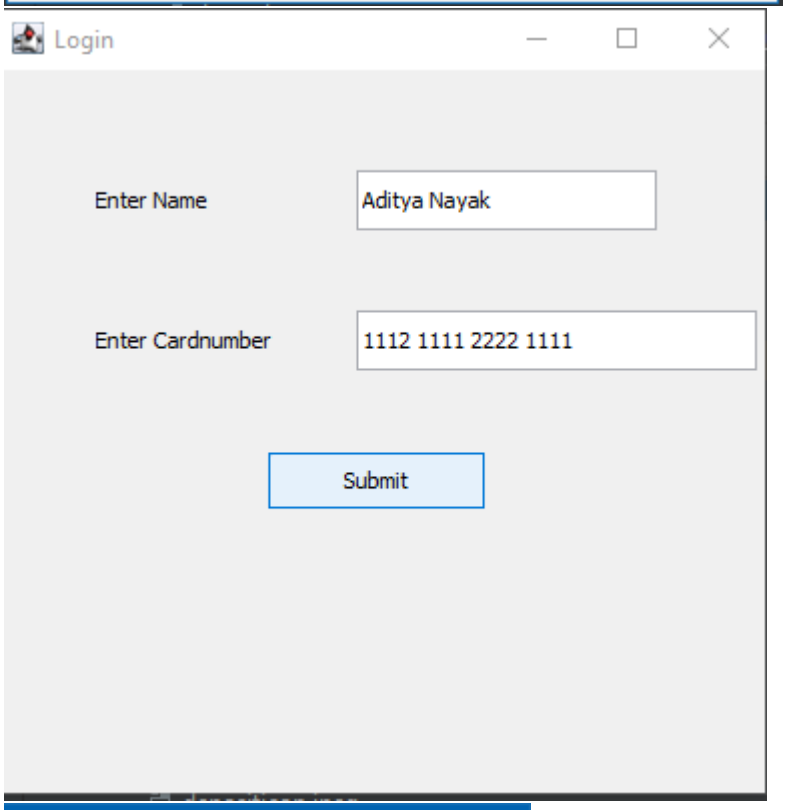
START PAGE
ATM
Insert your card
ATM CARD
A . M . A
N26

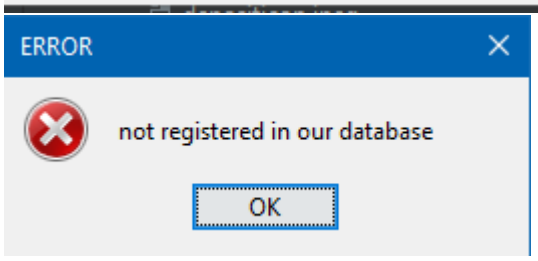

login signup
login
signup


signup
Enter Name
Enter Cardnumber
signup


ERROR
enter credentials
OK


signup
Enter Name
Raj Ule
Enter Cardnumber
0000 0000 0000 0001
signup


Login
Enter Name
Aditya Nayak
Enter Cardnumber
1112 1111 2222 1111
Submit


ERROR
not registered in our database
OK

the options we provide

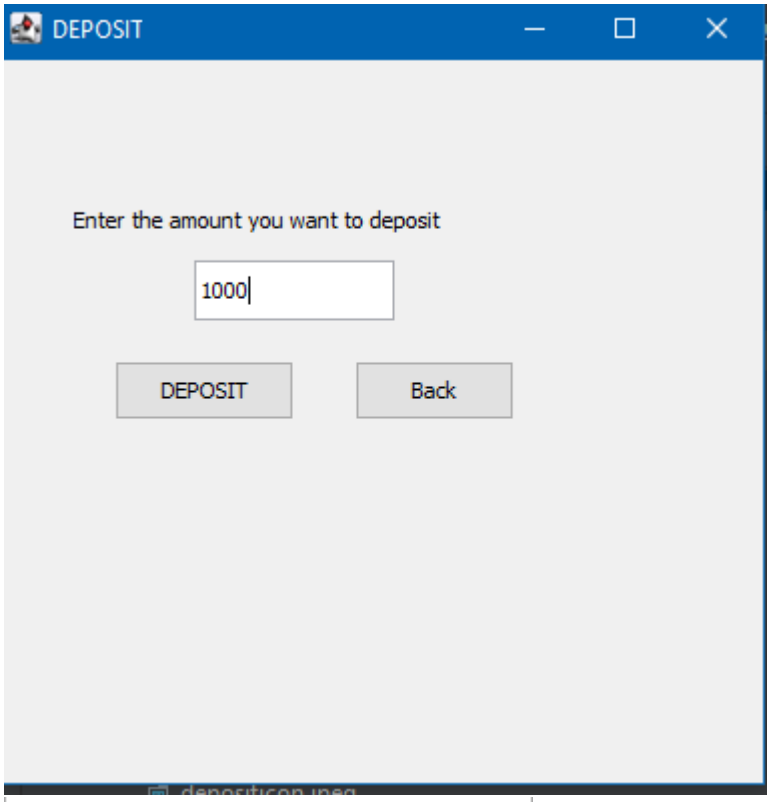
Deposit

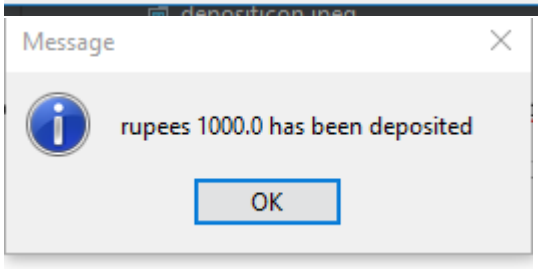
Withdraw

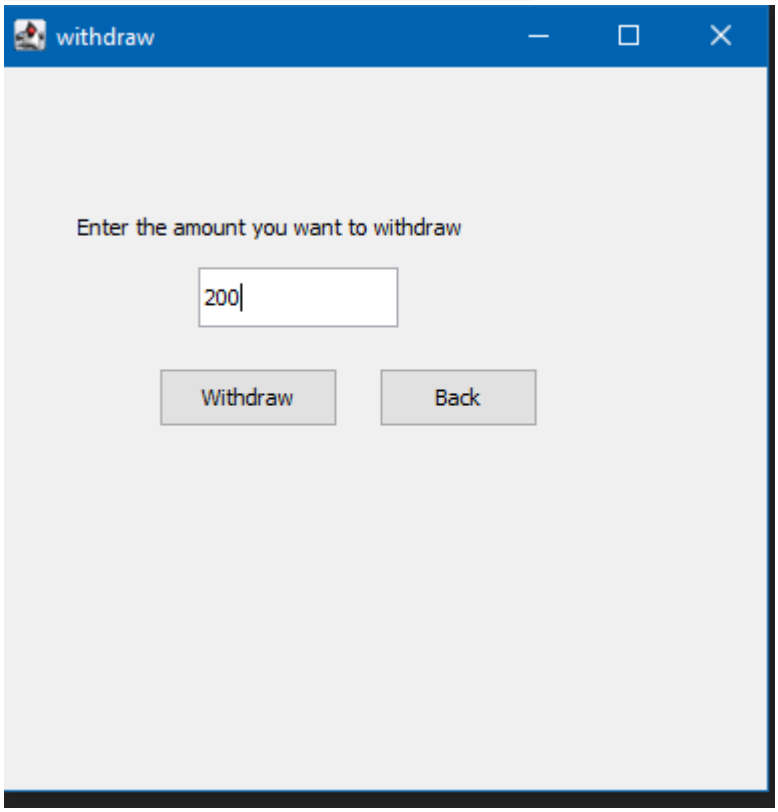
Balance

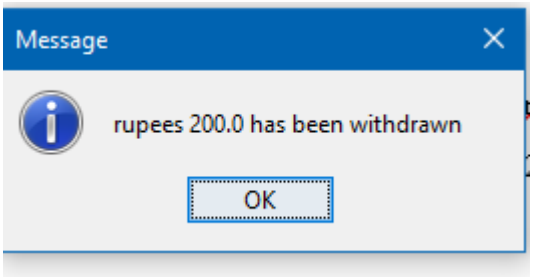
Receipt

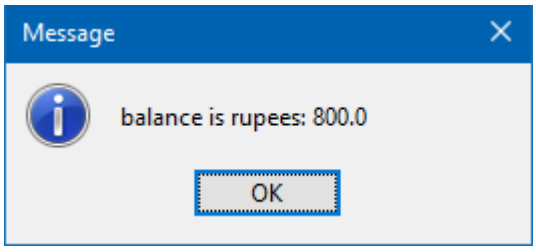
EXIT

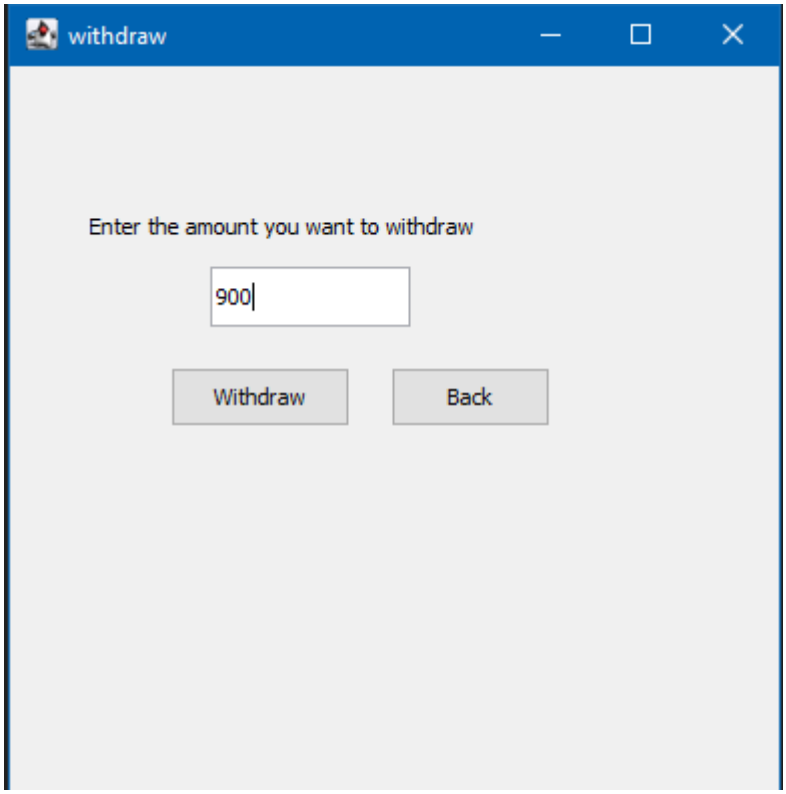

DEPOSIT
Enter the amount you want to deposit
1000
DEPOSIT
Back

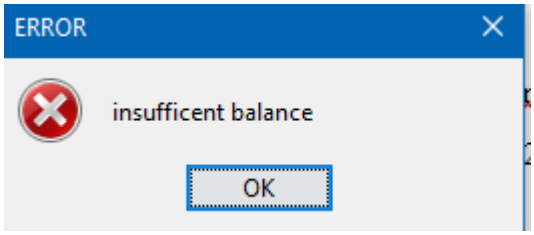

Message
rupees 1000.0 has been deposited
OK

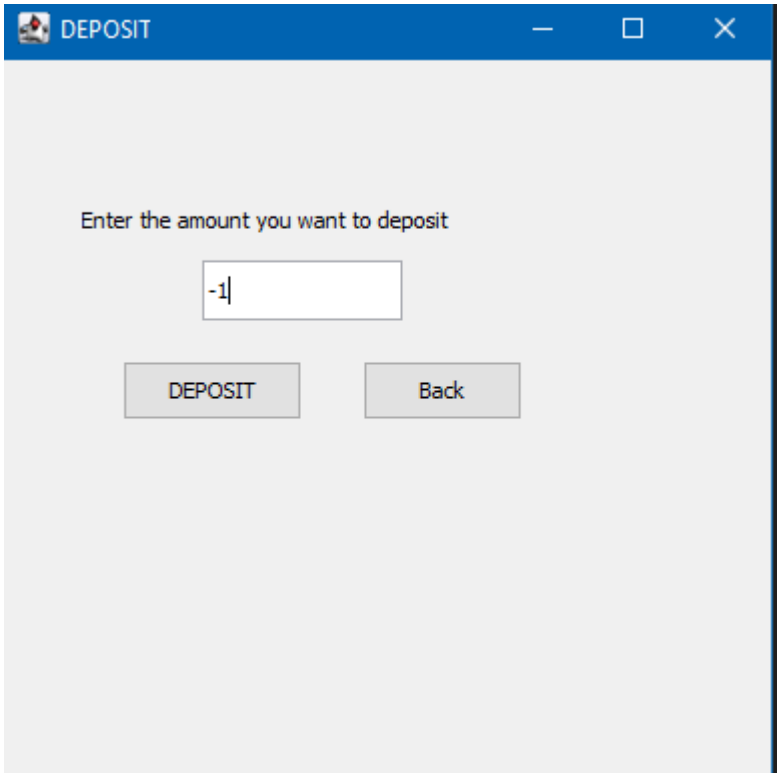

withdraw
Enter the amount you want to withdraw
200
Withdraw
Back

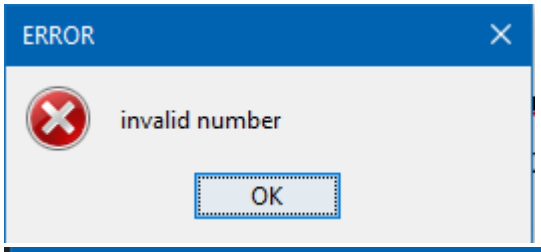

Message
rupees 200.0 has been withdrawn
OK

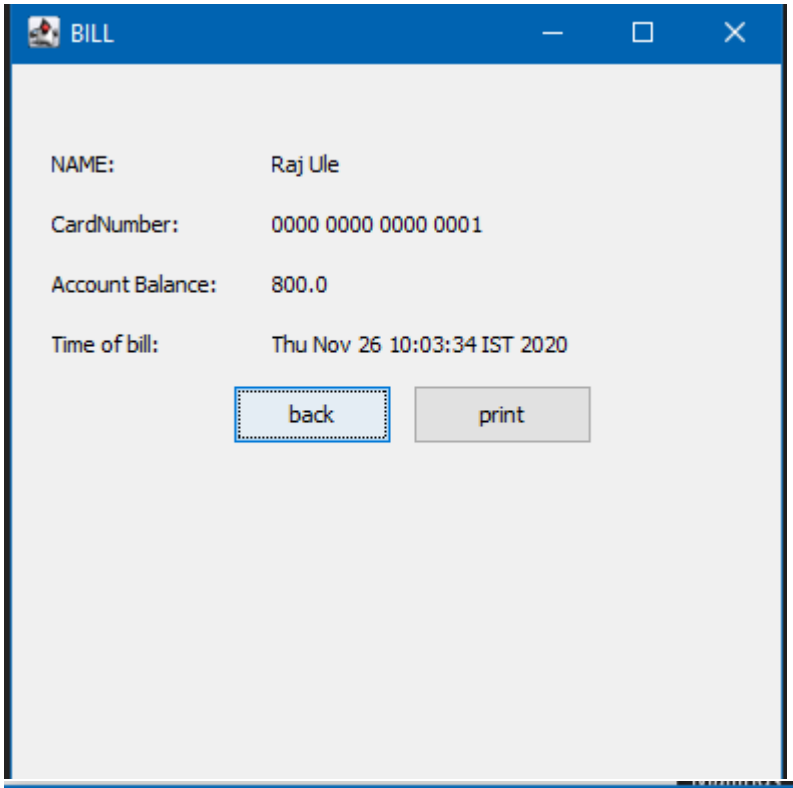

Message
balance is rupees: 800.0
OK

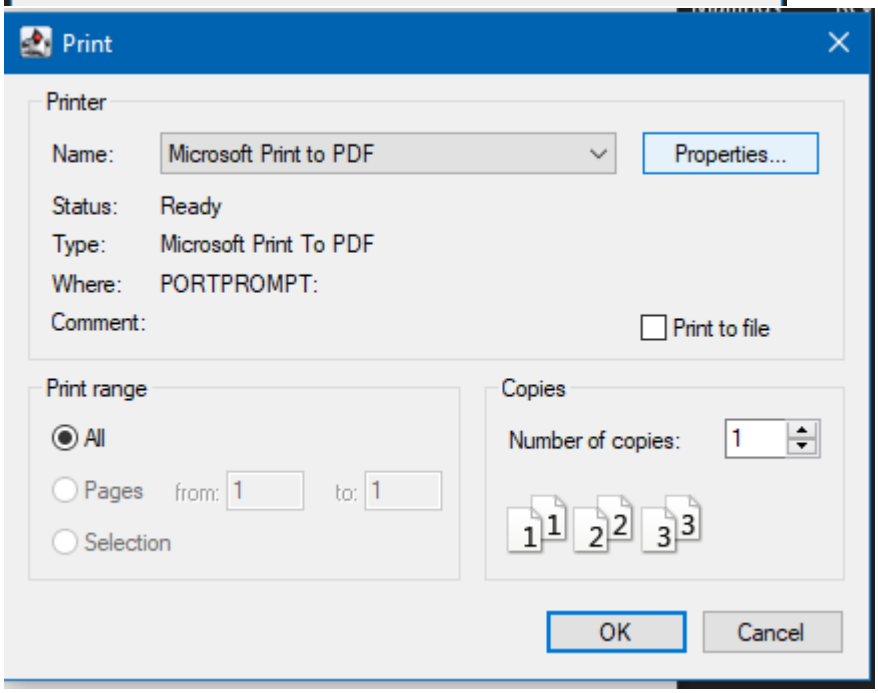

withdraw
Enter the amount you want to withdraw
900
Withdraw
Back

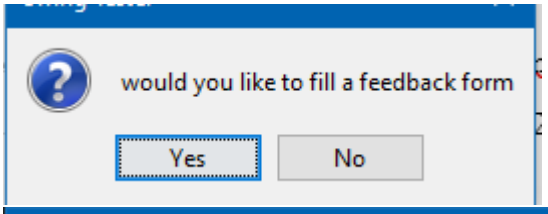

ERROR
insufficent balance
OK

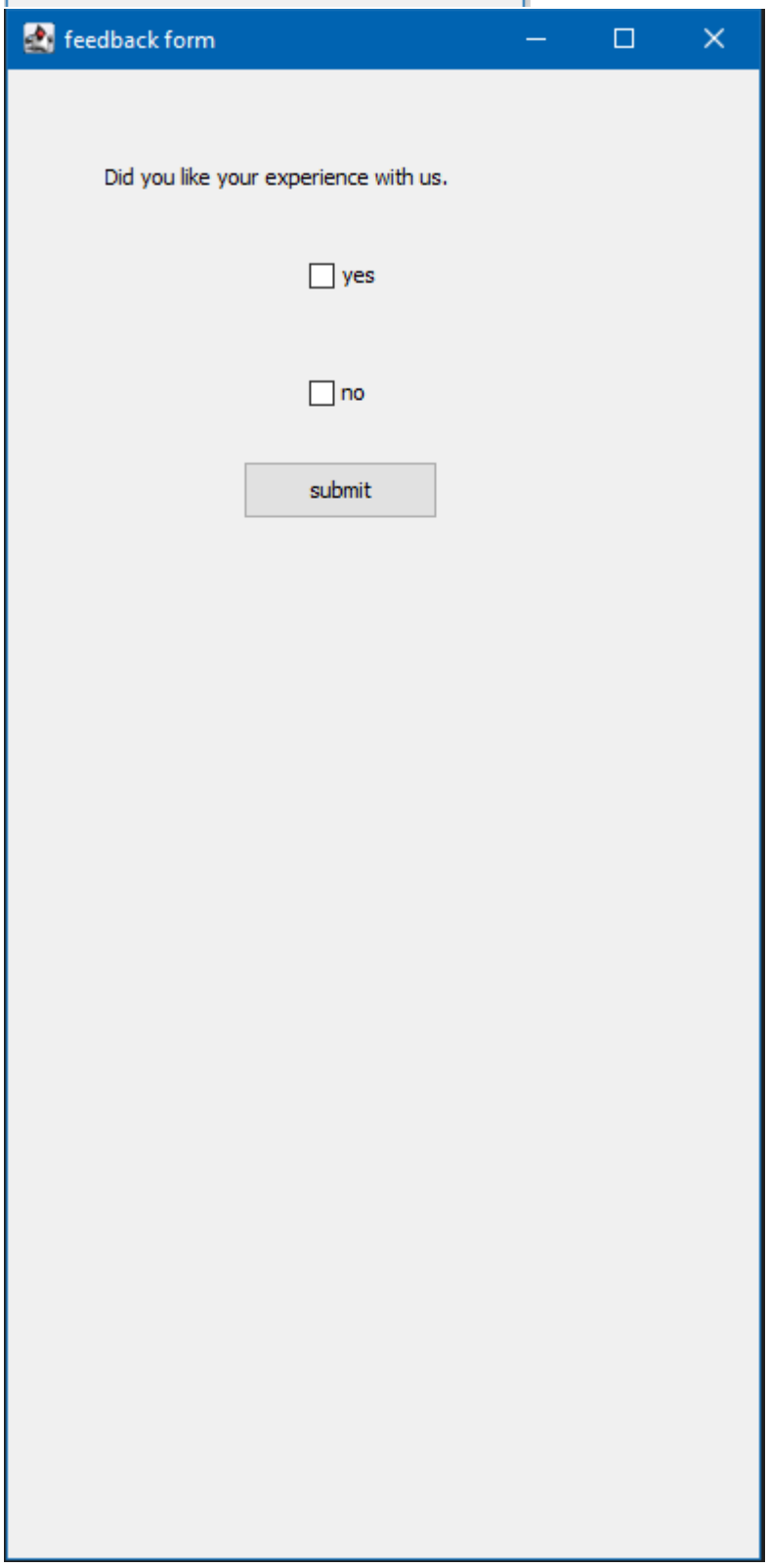

DEPOSIT
Enter the amount you want to deposit
-1
DEPOSIT
Back

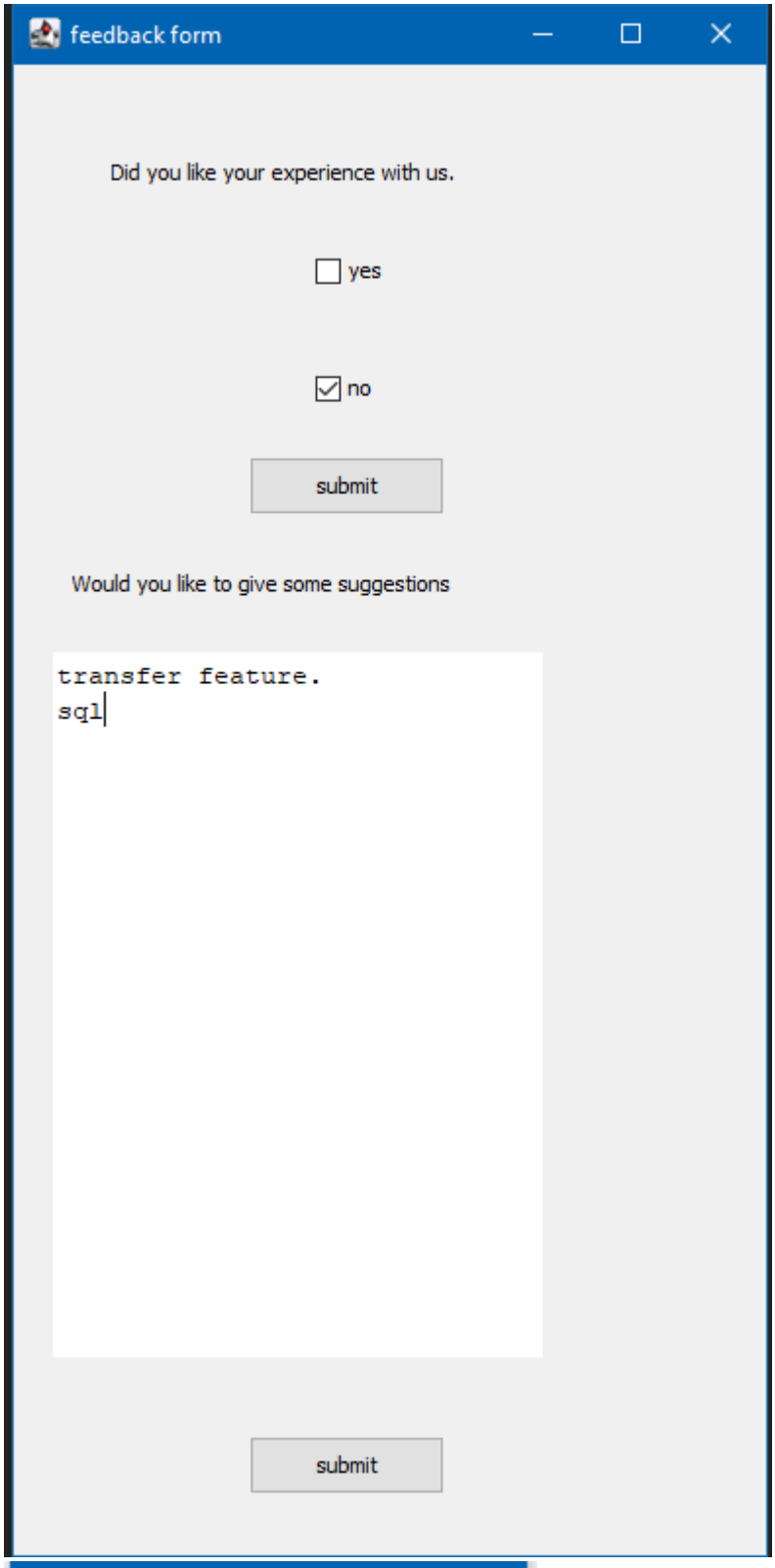

ERROR
invalid number
OK

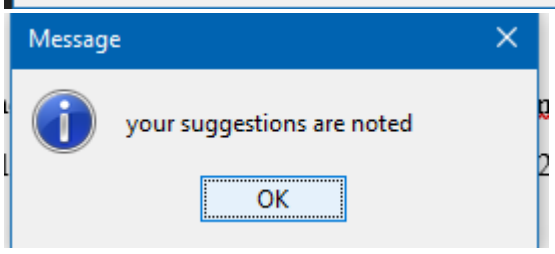

BILL
NAME:
Raj Ule
CardNumber:
0000 0000 0000 0001
Account Balance:
800.0
Time of bill:
Thu Nov 26 10:03:34 IST 2020
back
print

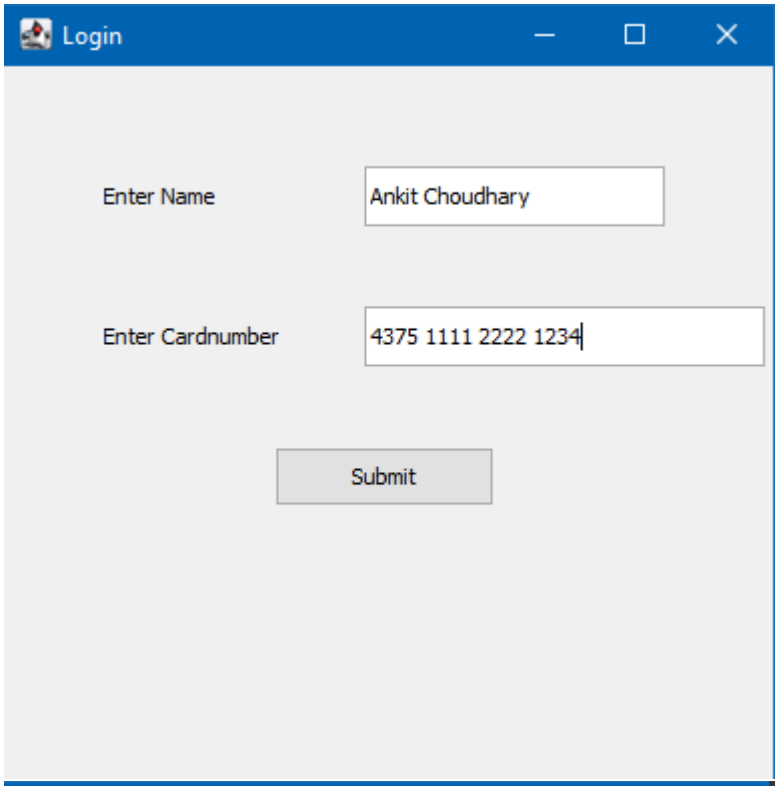

Print
Printer
Name:
Microsoft Print to PDF
Properties...
Status:
Ready
Type:
Microsoft Print To PDF
Where:
PORTPROMPT:
Comment:
Print to file
Print range
All
Pages
from:
1
to:
1
Selection
Copies
Number of copies:
1
OK
Cancel

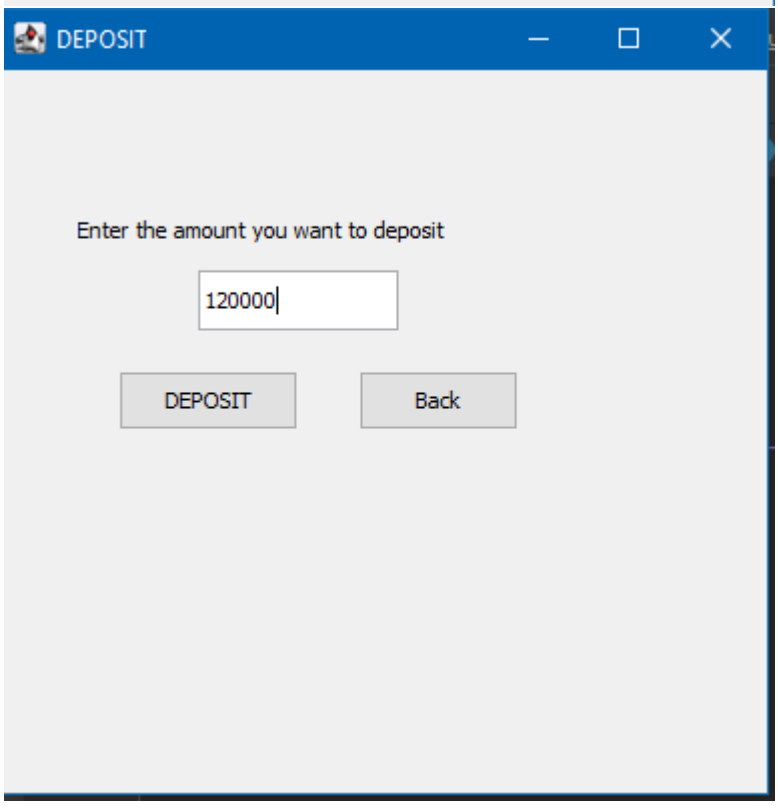

would you like to fill a feedback form
Yes
No

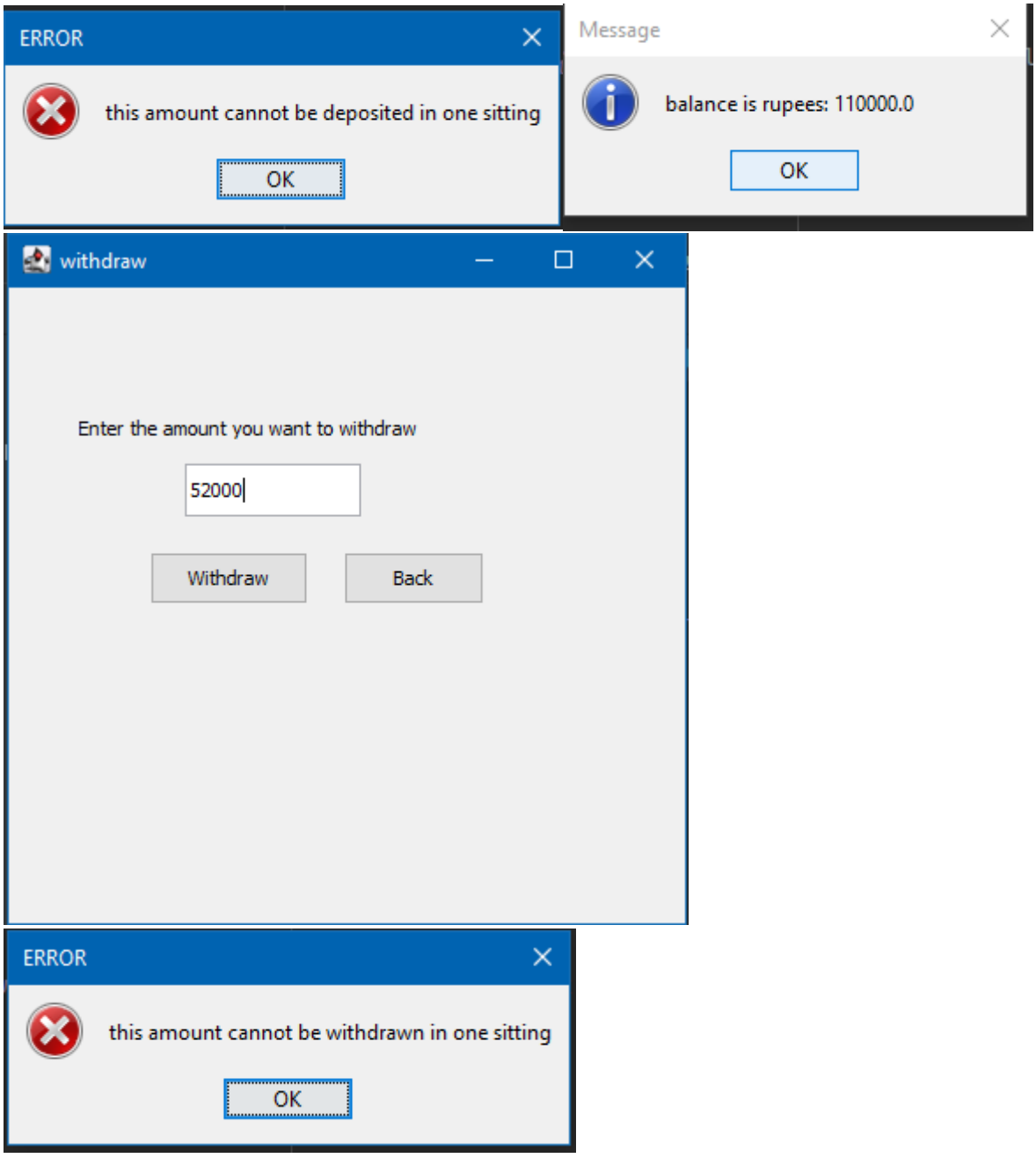

feedback form
Did you like your experience with us.
yes
no
submit

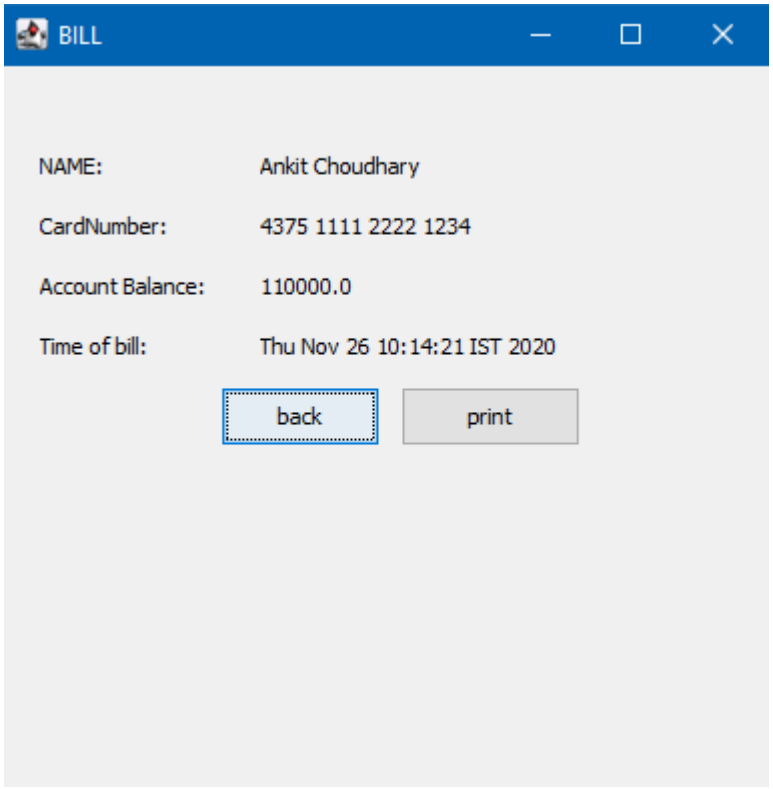

feedback form
Did you like your experience with us.
yes
no
submit
Would you like to give some suggestions
transfer feature.
sql
submit

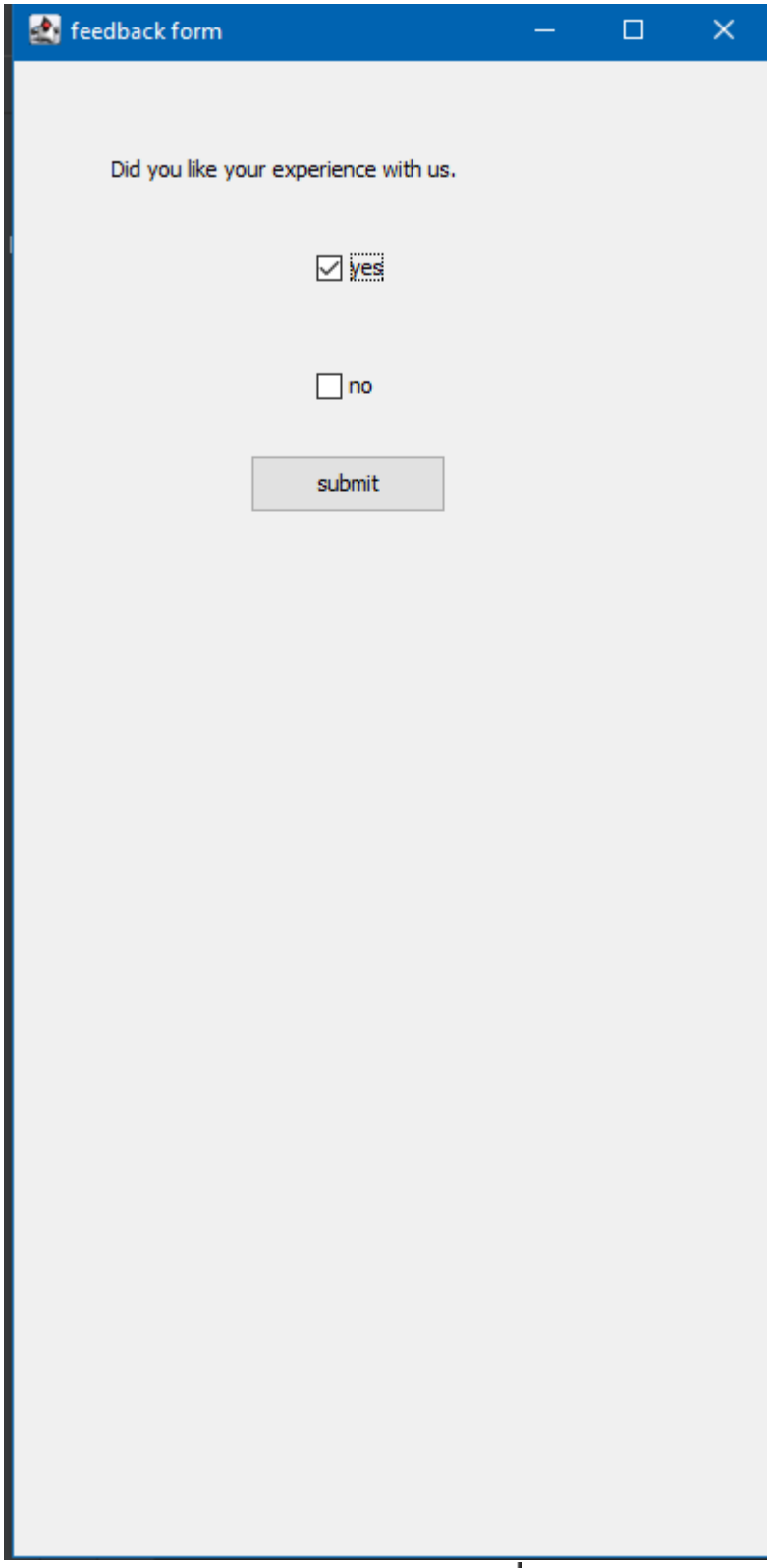

Message
your suggestions are noted
OK

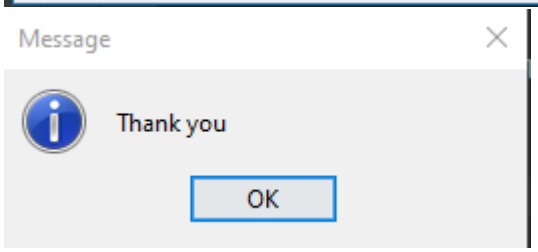

Login
Enter Name
Ankit Choudhary
Enter Cardnumber
4375 1111 2222 1234
Submit


DEPOSIT
Enter the amount you want to deposit
120000
DEPOSIT
Back


ERROR
this amount cannot be deposited in one sitting
OK
Message
balance is rupees: 110000.0
OK
withdraw
Enter the amount you want to withdraw
52000
Withdraw
Back
ERROR
this amount cannot be withdrawn in one sitting
OK


BILL
NAME:
Ankit Choudhary
CardNumber:
4375 1111 2222 1234
Account Balance:
110000.0
Time of bill:
Thu Nov 26 10:14:21 IST 2020
back
print


feedback form
Did you like your experience with us.
yes
no
submit


Message
Thank you
OK