新增功能

- 本程序是在I/O基础上完成的，一开始是想全部改成数据库的方式实现信息的存储，但是要改动老师的源码，所以就放弃了，新增功能还是将信息存储在本地的文档中，所以在data目录下新增了一个cards.txt文档
- 这个选课系统感觉不是单纯的选课系统，还能查询成绩，我索性将它弄成和教务系统差不多的东西，所以添加了和饭卡的相关的功能
- 此饭卡相关的功能针对学生,学生可以查询饭卡的余额和充值饭卡,管理员负责给未给饭卡开户的学生开户

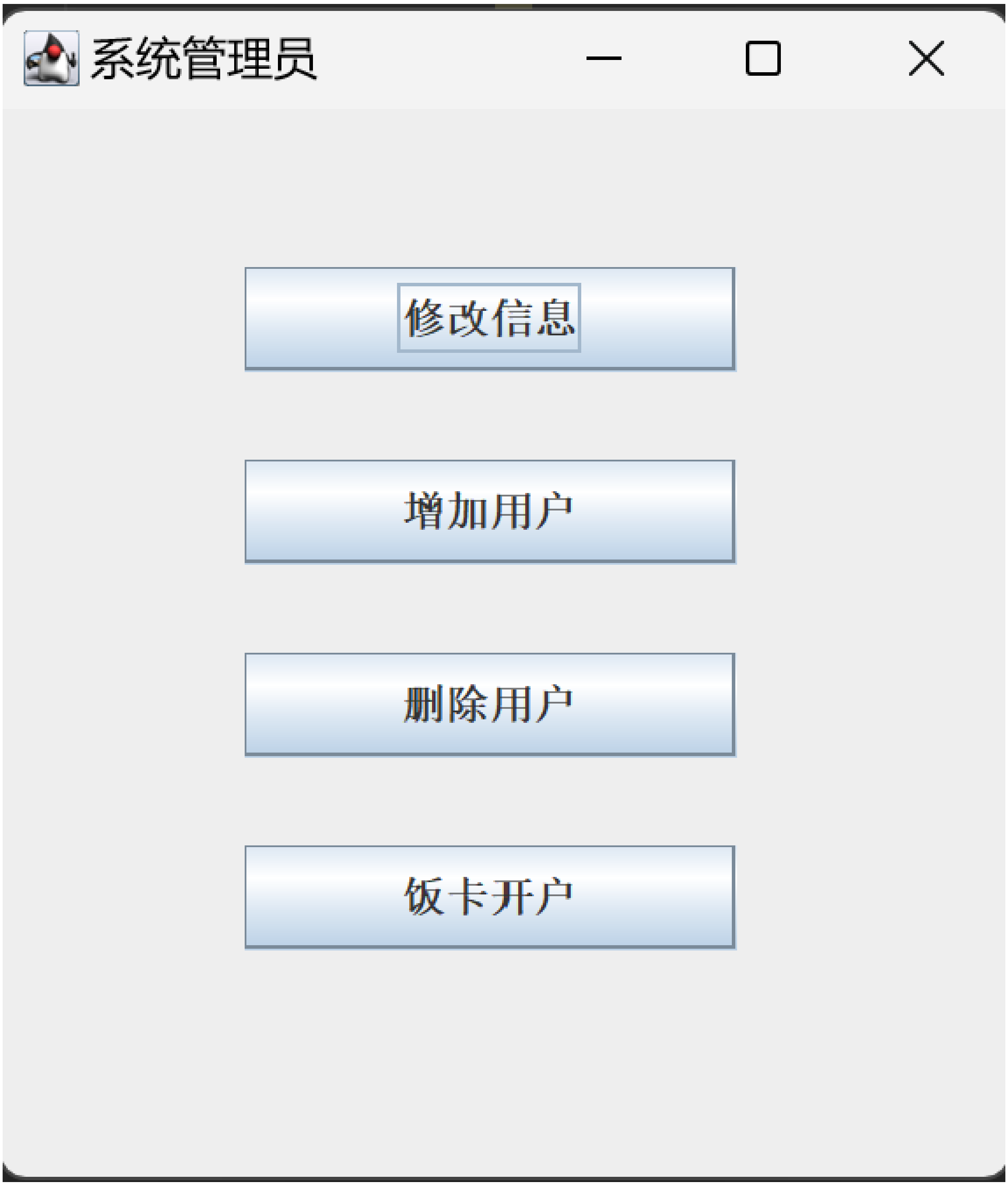
1.首先是管理人员可以给学生饭卡开户

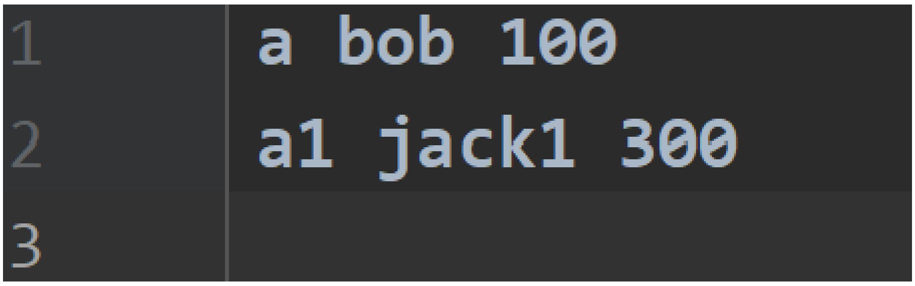
```java
package controller;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.io.*;
import java.util.ArrayList;

public class AddAccount extends JFrame implements ActionListener {
    JPanel contain;
    JLabel id,name,balance;
    JTextField idt,namet,balancet;
    JButton submit;
    public AddAccount(){
        super("add account");
        setSize(300,350);
        setLocation(600,400);
        contain = new JPanel();
        contain.setLayout(null);
        id = new JLabel("id");
        name = new JLabel("name");
        balance = new JLabel("balance");
        submit = new JButton("submit");
        idt = new JTextField();
        namet = new JTextField();
        balancet = new JTextField();
        id.setBounds(42, 20, 75, 35);
        idt.setBounds(80, 20, 150, 35);
        name.setBounds(42, 60, 75, 35);
        namet.setBounds(80, 60, 150, 35);
        balance.setBounds(42,100,150,35);
        balancet.setBounds(85,100,150,35);
        submit.setBounds(102, 260, 100, 30);
        contain.add(id);
        contain.add(idt);
        contain.add(name);
        contain.add(namet);
        contain.add(balance);
        contain.add(balancet);
        contain.add(submit);
        submit.addActionListener(this);
        add(contain);
        setVisible(true);
        enableEvents(AWTEvent.WINDOW_EVENT_MASK);
    }
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == submit) {
            if ((idt.getText().equals("")) || (namet.getText().equals("")) || (balancet.getText().equals(""))) {
                JOptionPane.showMessageDialog(null, "The message can not be empty! ", "tip", JOptionPane.INFORMATION_MESSAGE);
            } else {
                String file = System.getProperty("user.dir")+"/data/cards.txt";
                //首先判断是否该卡账户已经被创建,顺便将原有信息存储起来
                ArrayList<String> modifiedContent = new ArrayList<String>();
                try{
                    BufferedReader br = new BufferedReader(new FileReader(file));
                    String s =null;
                    while((s=br.readLine())!=null){
                        String[] result = s.split(" ");
                        if(result[0].equals(idt.getText()) && result[1].equals(namet.getText())){
                            JOptionPane.showMessageDialog(null, "This account is exist", "tip", JOptionPane.INFORMATION_MESSAGE);
                            return;
                        }
                        String s1 = "";
                        for (int i = 0; i < result.length - 1; i++) {
                            s1 = s1 + result[i];
                            s1 = s1 + " ";
                        }
                        s1 = s1 + result[result.length - 1];
                        modifiedContent.add(s1);
                    }
                    br.close();
                } catch (IOException ex) {
                    ex.printStackTrace();
                }
                String account = idt.getText() + " " + namet.getText() + " " + balancet.getText();
                modifiedContent.add(account);
                try {
                    FileWriter fw = new FileWriter(file);
                    BufferedWriter bw = new BufferedWriter(fw);

                    for (int i = 0; i < modifiedContent.size(); i++) {
                        bw.write(modifiedContent.get(i));
                        bw.newLine();
                    }
                    bw.close();
                    fw.close();
                } catch (IOException e1) {
                    e1.printStackTrace();
                }
                JOptionPane.showMessageDialog(null, "Create an account successfully", "tip", JOptionPane.INFORMATION_MESSAGE);
            }
        }
    }
    public void processWindowEvent(WindowEvent e) {
        if (e.getID() == WindowEvent.WINDOW_CLOSING) {
            this.dispose();
            setVisible(false);
        }
    }
}
```

下面演示它的功能:

我初始化card.txt文档时,已经写入了jack和jack1的饭卡,但是jack2还未开户,所以就给jack2开户

(1)进入到管理员界面,选择饭卡开户

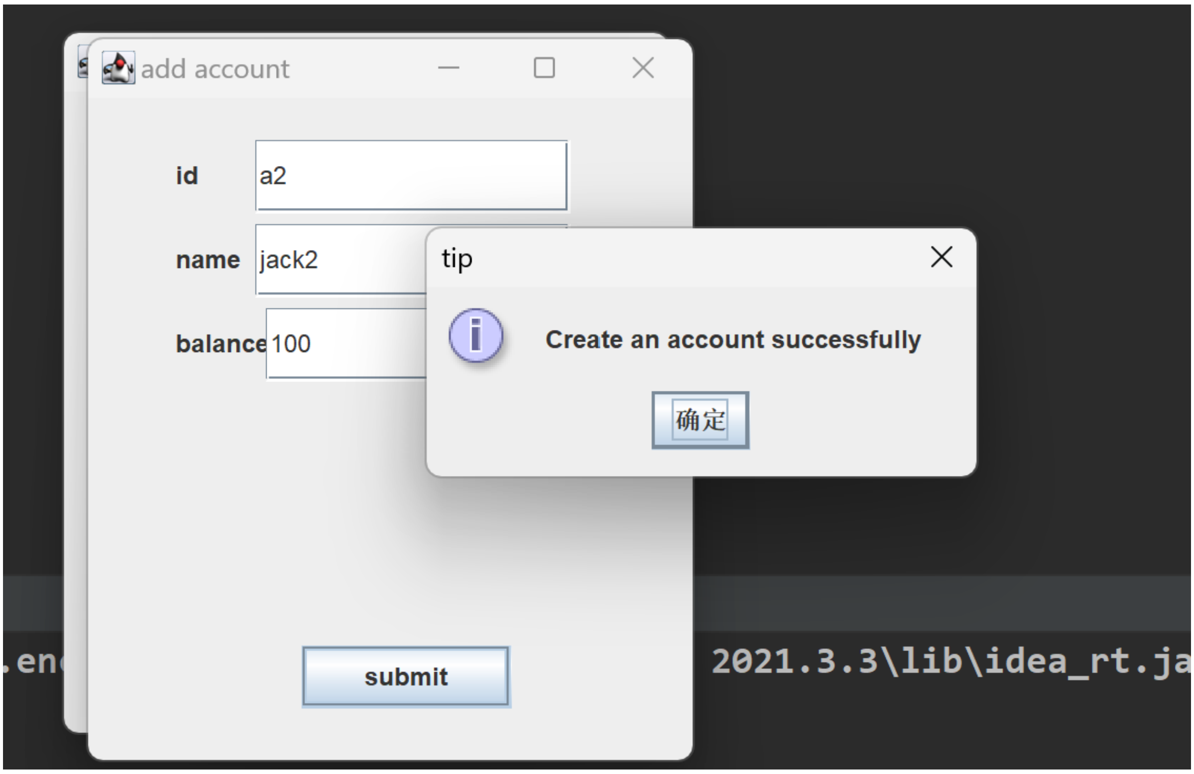




```
a bob 100
a1 jack1 300

```

此时的cards.txt里还没有jack2的饭卡信息

(2)给jack2开户



id a2

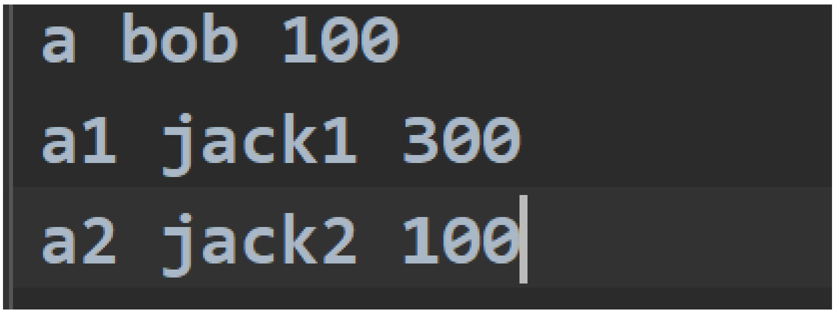
name jack2

balance 100

Create an account successfully

```
.en
2021.3.3\lib\idea_rt.ja
```

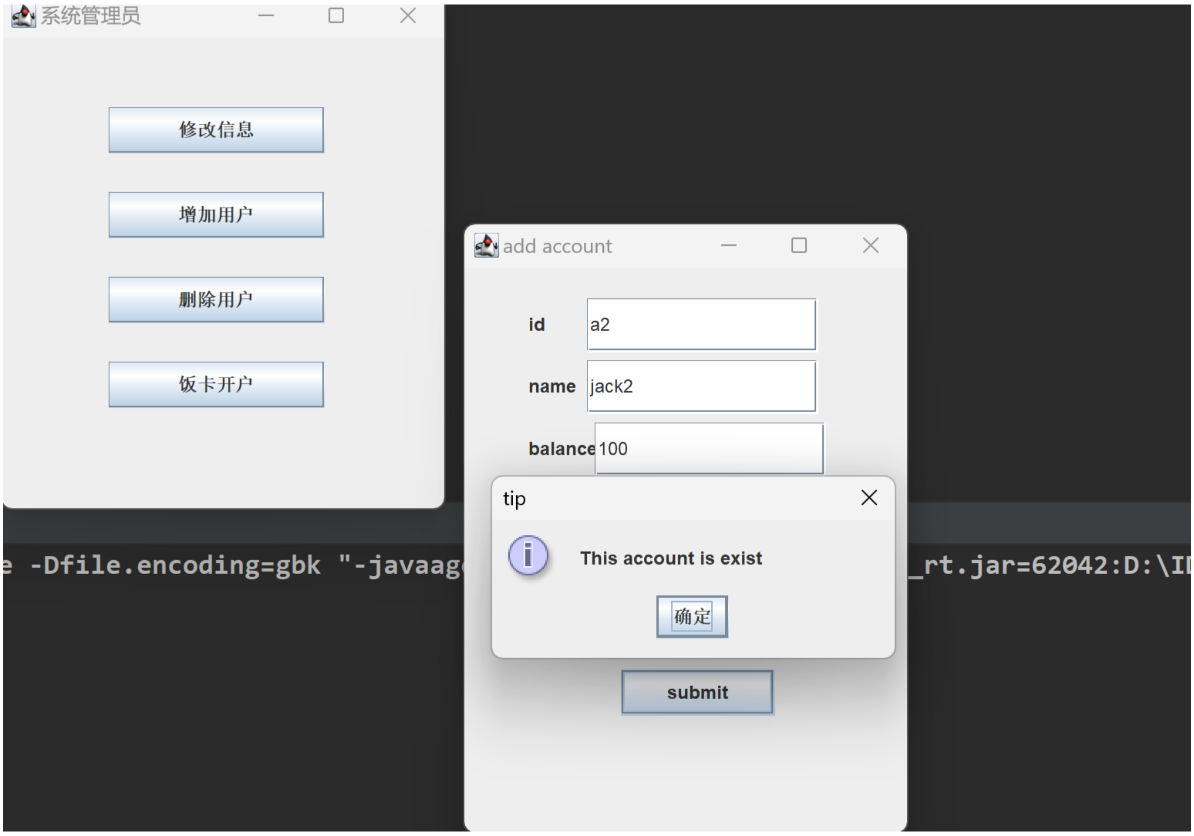
管理端显示开户成功,我们来看一下cards.txt里是否成功添加jack2的饭卡信息



```
a bob 100
a1 jack1 300
a2 jack2 100
```

成功添加!

(3)如果已经开户的学生还能开户吗?我们再拿jack2试一试



id a2

name jack2

balance 100

```
e -Dfile.encoding=gbk "-javaag                                         _rt.jar=62042:D:\II
```

This account is exist

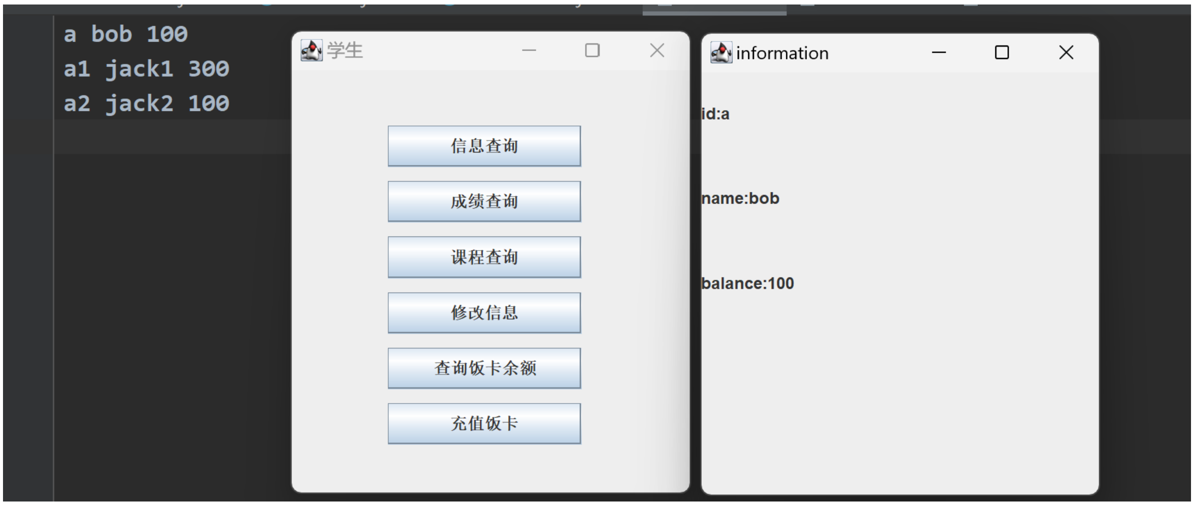
这时就会提示我们这个账户已经存在了(判断条件为id和name都一样才认为是同一个人,毕竟还是有重名的)

2.学生可以查询自己的饭卡余额

```java
package controller;

import javax.swing.*;
import java.awt.*;
import java.awt.event.WindowEvent;
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;

public class CardInfo extends JFrame {
    private static final long serialVersionUID = 1L;
    JLabel idLabel,nameLabel,balanceLabel;
    String id,name,balance;
    JPanel cardInfoJPanel;
    public CardInfo(String id){
        super("information");
        this.id = id;
        setSize(300,340);
        setLocation(600,400);
        cardInfoJPanel = new JPanel();
        cardInfoJPanel.setLayout(new GridLayout(5,1));
        add(cardInfoJPanel);
        String file = "";
        file = System.getProperty("user.dir")+"/data/cards.txt";
        try {
            BufferedReader bufferedReader = new BufferedReader(new FileReader(file));
            String s =null;
            while((s = bufferedReader.readLine())!=null) {
                String[] result = s.split(" ");
                if (result[0].equals(id)) {
                    id = result[0];
                    name = result[1];
                    balance = result[2];
                    idLabel = new JLabel("id:" + id);
                    nameLabel = new JLabel("name:" + name);
                    balanceLabel = new JLabel("balance:" + balance);
                }
            }
            bufferedReader.close();
            } catch (FileNotFoundException e) {
                e.printStackTrace();
            } catch (IOException e) {
                e.printStackTrace();
            }
        cardInfoJPanel.add(idLabel);
        cardInfoJPanel.add(nameLabel);
        cardInfoJPanel.add(balanceLabel);
        enableEvents(AWTEvent.WINDOW_EVENT_MASK);
        setVisible(true);
    }

    public void processWindowEvent(WindowEvent e){
        if(e.getID()==WindowEvent.WINDOW_CLOSING){
            this.dispose();
            setVisible(false);
        }
    }
}
```

下面演示它的功能:

(1)进入到学生端界面,选择查询饭卡余额(用id为a的学生为例)



```
a bob 100
a1 jack1 300
a2 jack2 100
```

information

id:a

name:bob

balance:100

可以看到和文档中的信息是符合的

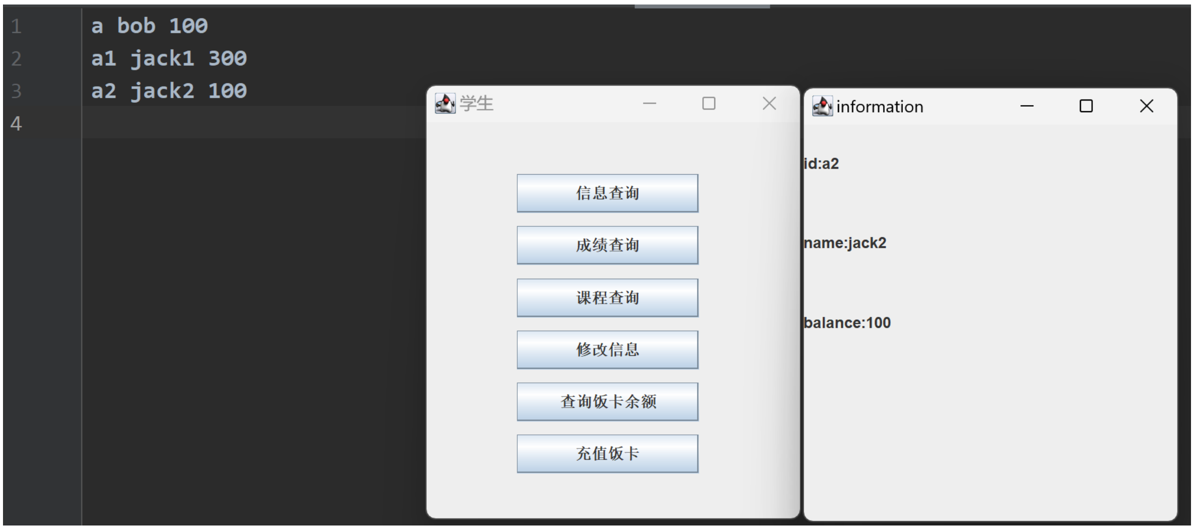
3.学生可以充值饭卡

```java
package controller;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowEvent;
import java.io.*;
import java.util.ArrayList;

public class Recharge extends JFrame implements ActionListener {
    String id;
    JButton submit;
    JPanel contain;
    JLabel money1, money2;
    JTextField money1t, money2t;

    public Recharge(String id) {
        super("Recharge card");
        setSize(300, 420);
        setLocation(600, 400);
        this.id = id;
        contain = new JPanel();
        contain.setLayout(null);
        money1 = new JLabel("money1");
        money2 = new JLabel("money2");
        submit = new JButton("submit");
        money1t = new JTextField();
        money2t = new JTextField();
        money1.setBounds(36, 100, 75, 35);
        money1t.setBounds(90, 100, 150, 35);
        money2.setBounds(36, 180, 75, 35);
        money2t.setBounds(90, 180, 150, 35);
        submit.setBounds(102, 300, 100, 30);
        contain.add(money1);
        contain.add(money1t);
        contain.add(money2);
        contain.add(money2t);
        contain.add(submit);
        submit.addActionListener(this);
        add(contain);
        setVisible(true);
        enableEvents(AWTEvent.WINDOW_EVENT_MASK);
    }

    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == submit) {
            if ((money1t.getText().equals("")) || (money2t.getText().equals(""))) {
                JOptionPane.showMessageDialog(null, "The input amount can not be empty", "tip",
                        JOptionPane.INFORMATION_MESSAGE);
            } else {
                if (!(money1t.getText().equals(money2t.getText()))) {
                    JOptionPane.showMessageDialog(null, "The two inputs are different!", "tip",
                            JOptionPane.INFORMATION_MESSAGE);
                }else {
                    ArrayList<String> modifiedContent = new ArrayList<String>();
                    String file = System.getProperty("user.dir")
                            + "/data/cards.txt";
                    try{
                        BufferedReader br = new BufferedReader(new FileReader(file));
                        String s =null;
                        while((s= br.readLine())!=null){
                            String[] result = s.split(" ");
                            if (result[0].equals(id)){
                                result[0] = result[0];
                                result[1] = result[1];
                                result[2] = result[2].replace(result[2],Integer.parseInt(result[2])+Integer.parseInt(money2t.getText())+"");
                            }
                            String s1 = "";
                            for (int i = 0; i < result.length-1; i++) {
                                s1 = s1 + result[i];
                                s1 = s1 + " ";
                            }
                            s1 = s1 + result[2];
                            modifiedContent.add(s1);
                        }
                        br.close();
                    } catch (IOException ex) {
                        ex.printStackTrace();
                    }

                    try {
                        FileWriter fw = new FileWriter(file);
                        BufferedWriter bw = new BufferedWriter(fw);

                        for (int i = 0; i < modifiedContent.size(); i++) {
                            bw.write(modifiedContent.get(i));
                            bw.newLine();
                        }

                        bw.close();
                        fw.close();
                    } catch (IOException e1) {
                        e1.printStackTrace();
                    }
                }
                JOptionPane.showMessageDialog(null, "recharge successfully", "message",
                        JOptionPane.INFORMATION_MESSAGE);
            }
        }
```

```java

    }
    public void processWindowEvent(WindowEvent e) {
        if (e.getID() == WindowEvent.WINDOW_CLOSING) {
            this.dispose();
            setVisible(false);
        }
    }
}
```

下面演示它的功能

(1)进入学生端界面,选择充值饭卡,(依然选择jack2作为演示对象)



```
a bob 100
a1 jack1 300
a2 jack2 100
```

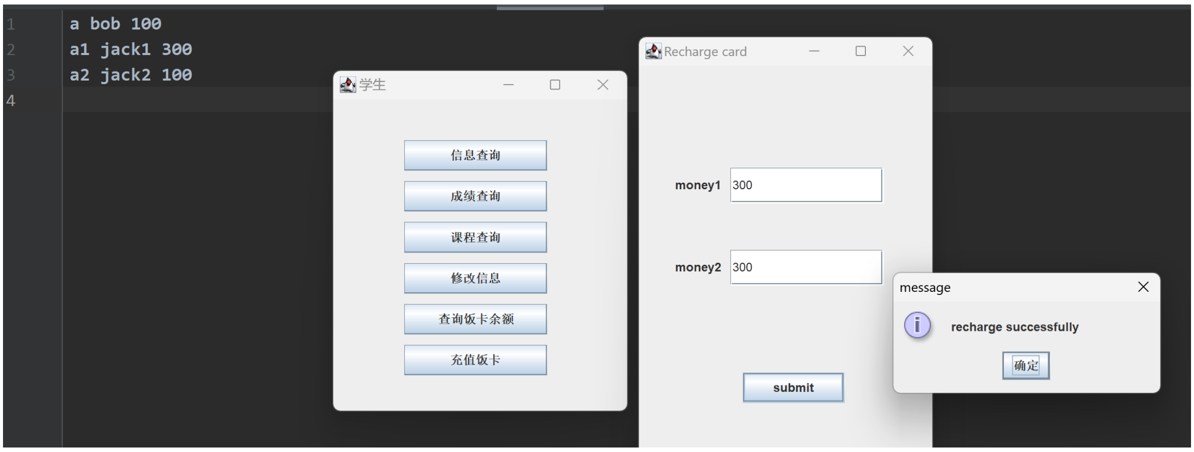
id:a2

name:jack2

balance:100

首先查询一下饭卡余额为100元

(2)选择充值饭卡,充值300元



```
a bob 100
a1 jack1 300
a2 jack2 100
```

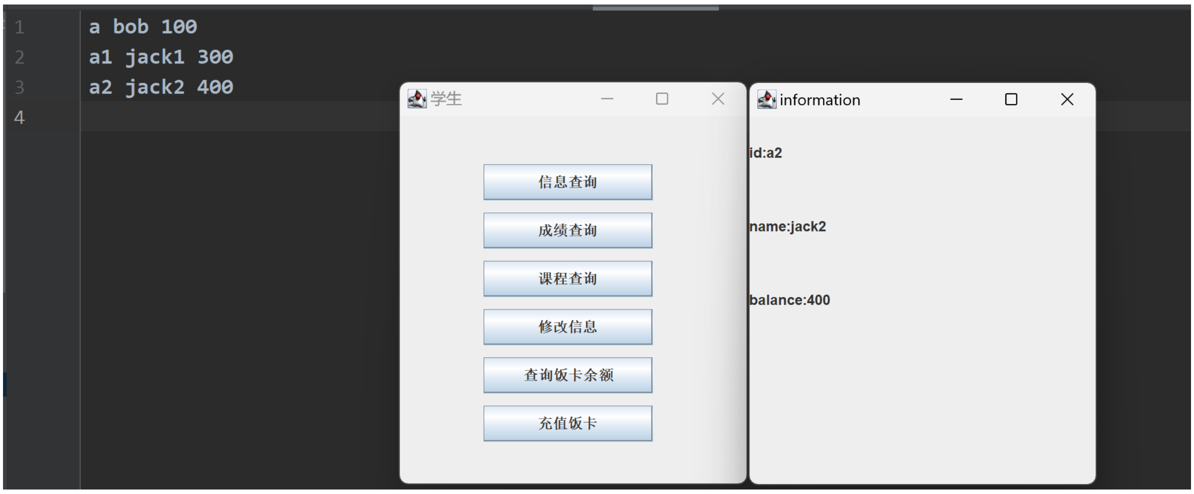
money1 300

money2 300

recharge successfully

可以看到提示充值成功

(3)再次查询饭卡余额



```
a bob 100
a1 jack1 300
a2 jack2 400
```

```
id:a2
name:jack2
balance:400
```

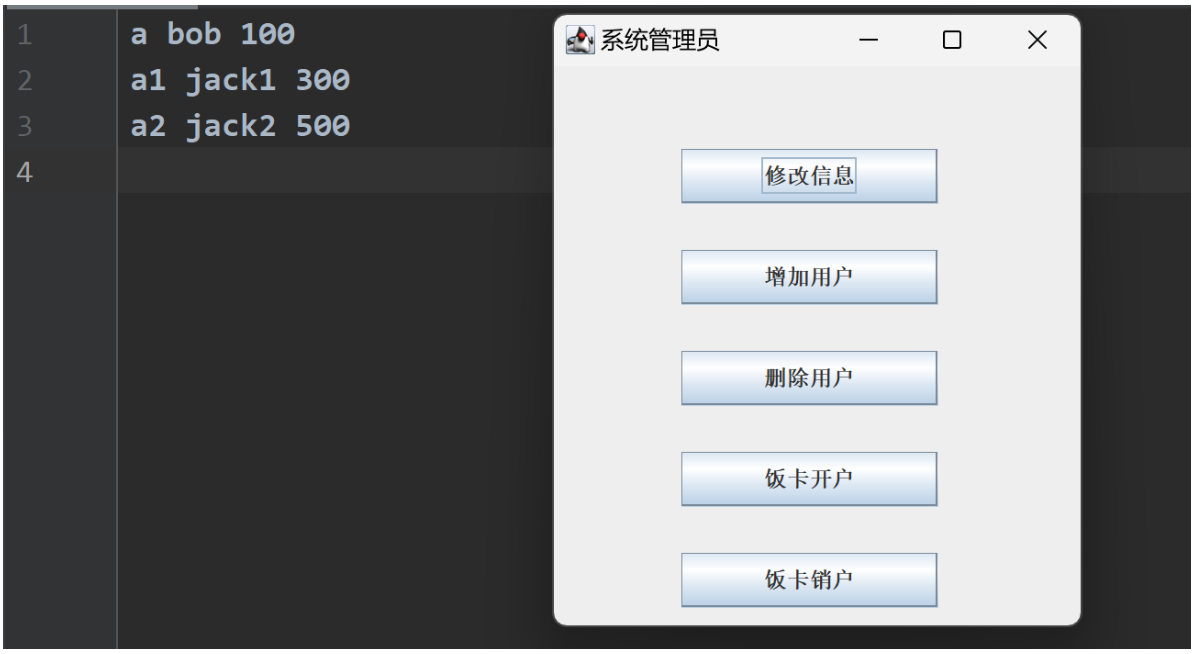
可以看到余额变成了400

4.管理员饭卡销户

```java
package controller;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.io.*;
import java.util.ArrayList;

public class DeleteAccount extends JFrame implements ActionListener {
    private static final long serialVersionUID = 1L;
    JPanel contain;
    JLabel id, name;
    JTextField idt, namet;
    JButton submit;

    public DeleteAccount() {
        super("delete account");
        setSize(300, 340);
        setLocation(600, 400);
        contain = new JPanel();
        contain.setLayout(null);
        id = new JLabel("id");
        name = new JLabel("name");
        submit = new JButton("submit");
        idt = new JTextField();
        namet = new JTextField();
        id.setBounds(42, 20, 75, 35);
        idt.setBounds(80, 20, 150, 35);
        name.setBounds(42, 60, 75, 35);
        namet.setBounds(80, 60, 150, 35);
        submit.setBounds(102, 150, 100, 30);
        contain.add(id);
        contain.add(idt);
        contain.add(name);
        contain.add(namet);
        contain.add(submit);
        submit.addActionListener(this);
        add(contain);
        setVisible(true);
        enableEvents(AWTEvent.WINDOW_EVENT_MASK);
    }

    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == submit) {
            if ((idt.getText().equals("")) || (namet.getText().equals(""))) {
                JOptionPane.showMessageDialog(null, "The message can not be empty! ", "tip", JOptionPane.INFORMATION_MESSAGE);
            } else {
                String file = System.getProperty("user.dir")+"/data/cards.txt";
                ArrayList<String> modifiedContent = new ArrayList<String>();
                int count=0;
                try{
                    BufferedReader br = new BufferedReader(new FileReader(file));
                    String s =null;

                    while((s=br.readLine())!=null){
                        String[] result1 = s.split(" ");
                        if(result1[0].equals(idt.getText()) && result1[1].equals(namet.getText())){
                            count=1;
                            break;
                        }else {
                            count=-1;
                        }
                    }

                } catch (IOException ex) {
                    ex.printStackTrace();
                }
                if(count==-1){
                    JOptionPane.showMessageDialog(null, "This account is not exist!", "tip", JOptionPane.INFORMATION_MESSAGE);
                }else {
                    try {
                    String n = null;
                    BufferedReader br = new BufferedReader(new FileReader(file));
                    while((n=br.readLine())!=null){
                        String[] result = n.split(" ");
                        if(!(result[0].equals(idt.getText()) && result[1].equals(namet.getText()))){
                            String s1 = "";
                            for (int i = 0; i < result.length - 1; i++) {
                                s1 = s1 + result[i];
                                s1 = s1 + " ";
                            }
                            s1 = s1 + result[result.length - 1];
                            modifiedContent.add(s1);
                        }
                    }
                    br.close();

                    FileWriter fw = new FileWriter(file);
                    BufferedWriter bw = new BufferedWriter(fw);

                    for (int i = 0; i < modifiedContent.size(); i++) {
                        bw.write(modifiedContent.get(i));
                        bw.newLine();
                    }

                    bw.close();
                    fw.close();
                    } catch (IOException e1) {
                        e1.printStackTrace();
                    }


                    JOptionPane.showMessageDialog(null, "Delete Account successfully", "tip", JOptionPane.INFORMATION_MESSAGE);
                }
            }
        }
    }
}
```

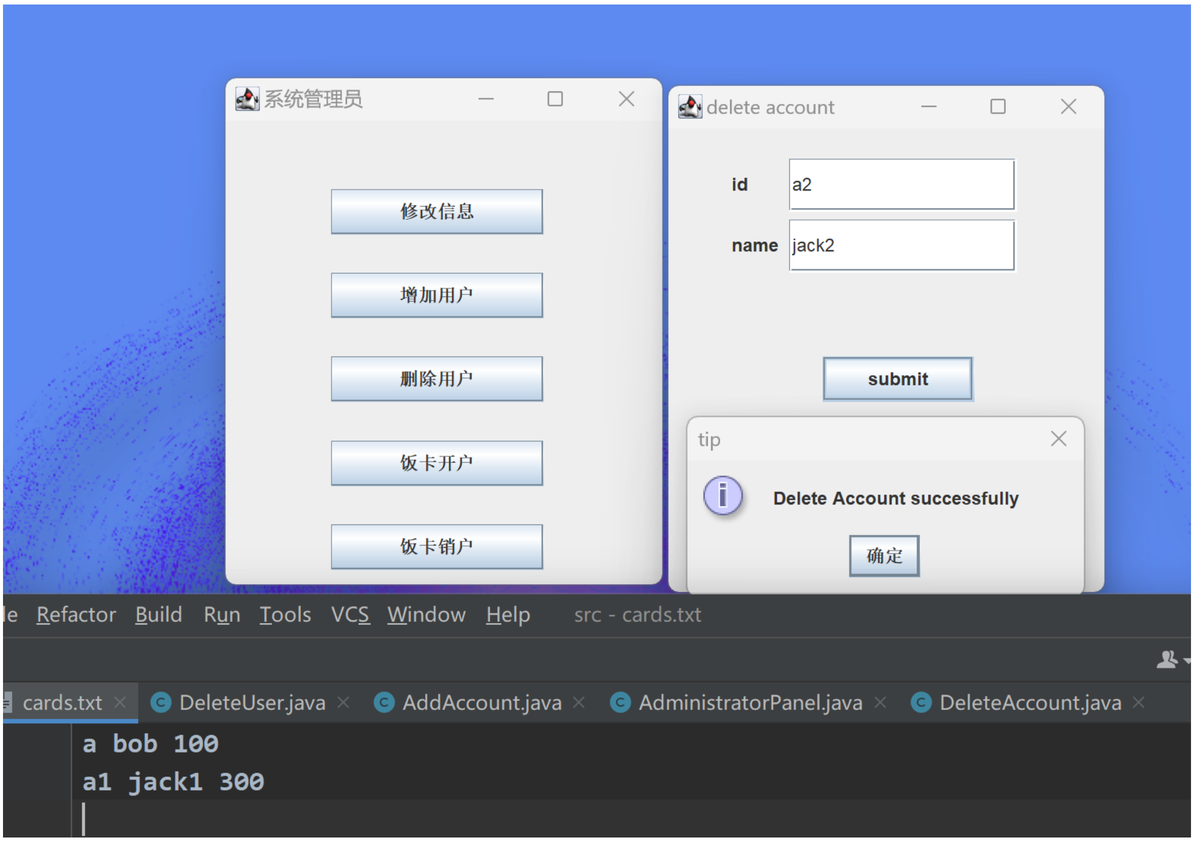
下面进行功能演示:

(1)进入管理端选择饭卡销户功能(以再cards.txt文档中存在的jack2为例)



```
a bob 100
a1 jack1 300
a2 jack2 500
```

(2)输入对应的id和name进行销户



id: a2

name: jack2

Delete Account successfully

```
a bob 100
a1 jack1 300
```

可以看到jack2的账户被删除掉了

总结

- 首先是饭卡功能还可以扩展很多,可以和支付相联系,但是这样必然会大改代码
- 其次这个功能相对简单,由我一个人完成
- 有很多想法还未实现,奈何技术没到位