

学生成绩管理系统

1 设计目的及内容要求

1、设计目的：

巩固和加深学生对高级语言程序设计课程的基本知识的理解和掌握，掌握 java 语言编程和程序调试的基本技能，利用 java 语言进行基本的软件设计，提高运用 java 语言解决实际问题的能力。

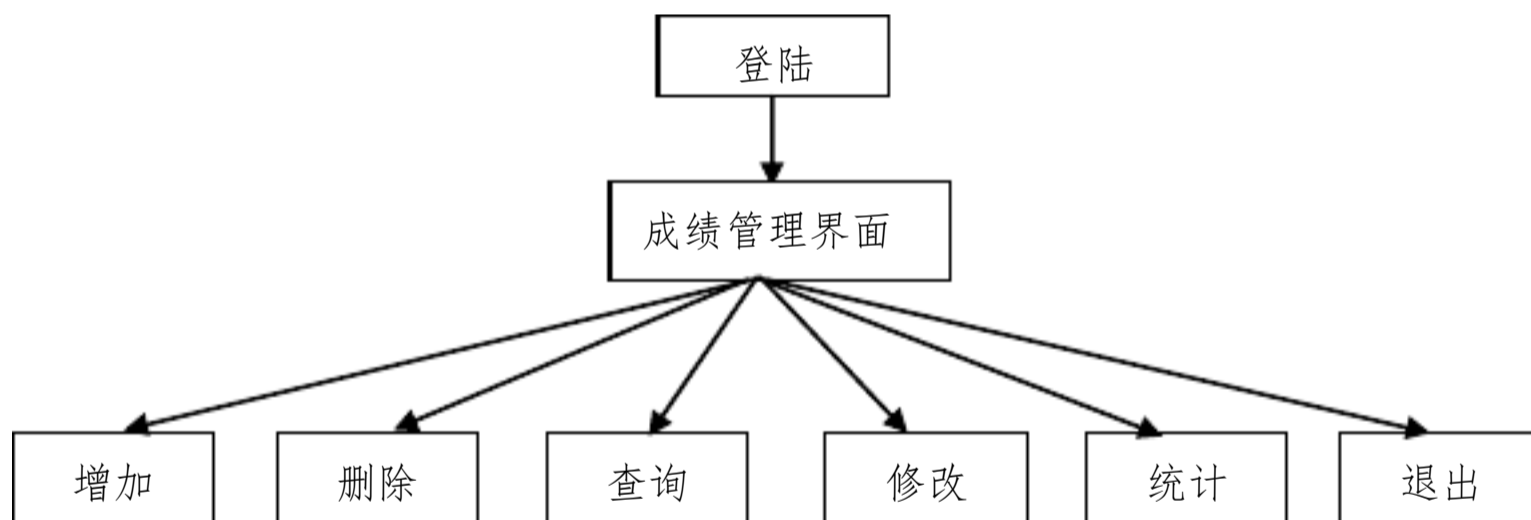
2、内容要求

实现学生成绩的管理（增、删、改、查询、持久化、成绩排序、成绩统计等功能），在文件中增加、删除、学生信息，根据学号查询、修改学生信息，统计功能求每个人的总分并按从高到低排序，通过这些操作对文件中的信息保存。

2 系统总体设计

成功登陆系统以后，出现成绩管理界面，系统初始化，可对学生成绩进行增加、删除、查询、修改、统计，进入相应界面进行成绩管理，退出系统自动保存本次操作内容，保存信息。

简易流程图：



3 系统详细设计

①

创建 **Student** 类，设置变量及对应方法

Student.java 代码：

```
package keshe;
import java.io.Serializable;
public class Student implements Serializable{
    private String name ;
    private int num;
    private int yuwen ;
    private int shuxue ;
    private int java ;
    private int sum=0;
    public Student(){
    }
    public Student(String name, int num, int yuwen, int shuxue,
        int java) {
        super ();
        this . name = name;
        this . num = num;
        this . yuwen = yuwen;
        this . shuxue = shuxue;
        this . java = java;
    }
    public String getName() {
        return name ;
    }
    public void setName(String name) {
        this . name = name;
    }
    public int getNum() {
        return num;
    }
    public void setNum( int num) {
        this . num = num;
    }
    public int getYuwen() {
        return yuwen ;
    }
    public void setYuwen( int yuwen) {
        this . yuwen = yuwen;
    }
}
```

```
}
public int getShuxue() {
    return shuxue ;
}
public void setShuxue( int shuxue) {
    this . shuxue = shuxue;
}
public int getJava() {
    return java ;
}
public void setJava( int java) {
    this . java = java;
}
public int getSum() {
    return sum ;
}
public void setSum( int sum) {
    this . sum = sum;
}
public String toString() {
    return          + name +          + num +          + yuwen
                    +          + shuxue +          + java +          + sum
                    +          ;
}
}
```

②

Student 管理类 **StuC**，创建对应方法，实现对 **Student** 对象的操作（增、删、改、查询、持久化、成绩排序、成绩统计等功能）供其对象调用。

StuC.java 代码

```
package keshe;
```

```
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.FileReader;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.io.ObjectOutputStream;
import java.util.ArrayList;
```

```
public class StuC {
```

```
ArrayList <Student> al=new ArrayList<Student>();

/添加
public void adds(Student s){

    al.add(s);
}

/删除
public void del(int n){
    for(int i=0;i<al.size();i++){
        if(al.get(i).getNum()==n)
            al.remove(i);
    }
}

/求总分
public void sum(){
    for(int i=0;i<al.size();i++){
        al.get(i).setSum(al.get(i).getJava()+al.get(i).getShuxue()+al.get(i).getYuwen());
    }
}

/排序
public void sort() {
    for (int i = 0; i < al.size(); i++) {
        for (int j = 0; j < al.size()-1-i; j++) {
            if (al.get(j).getSum() < al.get(j+1).getSum()) {
                Object o=al.get(j);
                al.set(j, al.get(j+1));
                al.set(j+1, (Student) o);
            }
        }
    }
}

public void paint(){
    for(int i=0;i<al.size();i++){
        System.out.println(al.get(i));
    }
}

public String toString() {

}

/输出流
public void stor()
{
```

```
ObjectOutputStream out = null;
try {
    out = new ObjectOutputStream(new FileOutputStream(file));
    out.writeObject(al);
    out.close();
} catch (FileNotFoundException e) {

    e.printStackTrace();
} catch (IOException e) {

    e.printStackTrace();
}

}
/输入流
public void read()
{
    ObjectInputStream in = null;
    try {
        in = new ObjectInputStream(new FileInputStream(file));
        try {
            al = (ArrayList<Student>)in.readObject();
        } catch (ClassNotFoundException e) {

            al=null;
        }
        in.close();
    } catch (FileNotFoundException e) {

    } catch (IOException e) {

        e.printStackTrace();
    }
}
/查找
public Student find(int n)
{
    for(int i=0;i<al.size();i++){
        if(al.get(i).getNum()==n){
            return al.get(i);
        }
    }
}
```

```
        return null;
    }
}
```

③

登陆界面及成绩管理的所有界面，同过创建 StuC 的对象，调用其方法实现成绩管理代码：

```
package keshe;
import java.awt.*;
import java.awt.event.*;
import java.io.FileNotFoundException;
import java.io.IOException;

import javax.swing.*;

public class Login extends JFrame {
    private TextField f1;
    private TextField f2;
    private JButton b1;
    private JButton b2;
    private JButton b3;
    StuC scs=new StuC();
    /登陆界面
    public Login(){
        Container cp=getContentPane();//容器
        cp.setLayout(new GridLayout(3,1));//三行一列布局
                用户名
                密 码

        Panel p1=new Panel();
        Panel p2=new Panel();
        Panel p3=new Panel();
        f1=new TextField(10);
        f2=new TextField(10);
        f2.setEchoChar('*');//回显字符为*
                登录
                重置
                退出

        p1.add(l1);//第一行添加 label 1
        p1.add(f1);
        p2.add(l2);
        p2.add(f2);
        p3.add(b1);
        p3.add(b2);
        p3.add(b3);
    }
}
```

```
cp.add(p1);
cp.add(p2);
cp.add(p3);
b1.addActionListener(new Enter());
b2.addActionListener(new ReWrite());
b3.addActionListener(new Close());
}
class Enter implements ActionListener{
    public void actionPerformed(ActionEvent e)
    {
        {
            scs.read();//初始化, 从文件读入信息
            XueSheng frame1 = new XueSheng();
            frame1.setBounds(200, 200, 300, 300);
            frame1.setVisible(true);
        }
        用户名或密码错误, 请重新登录!
    }
}
class ReWrite implements ActionListener{
    public void actionPerformed(ActionEvent e)
    {
        f1.requestFocus();
    }
}
class Close implements ActionListener{
    public void actionPerformed(ActionEvent e)
    {
        JButton bt=(JButton)e.getSource();
        if(bt==b3){
            System.exit(0);
        }
    }
}
/主函数 程序开始
public static void main(String[] args) {
    Login log=new Login();
        系统登录
    log.setBounds(200, 200, 300, 300);
    log.setBackground(Color.blue);
```

```
log.setVisible(true);
}
/信息管理界面内部类 进行初始化和保存
class XueSheng extends JFrame implements ActionListener {
    private JButton cx, zj, tc, sc,xg,tj;
    public XueSheng()
    {
        Container c = this.getContentPane();
        c.setLayout(new GridLayout(3, 1));

        JPanel panel2 = new JPanel();
        JPanel panel1 = new JPanel();
                                欢迎进入成绩管理

        label1.setForeground(Color.blue);
        c.add(label1);
        /添加按钮
                                查询
        panel2.add(cx);
                                增加
        panel2.add(zj);
                                删除
        panel2.add(sc);
                                退出
        panel2.add(tc);
                                修改
        panel1.add(xg);
                                统计
        panel1.add(tj);
        c.add(panel2);
        c.add(panel1);
        cx.addActionListener(this);
        zj.addActionListener(this);
        sc.addActionListener(this);
        xg.addActionListener(this);
        tc.addActionListener(this);
        tj.addActionListener(this);
        this.setVisible(true);
    }
    public void actionPerformed(ActionEvent e)
    {
        if (e.getSource() == cx) {
            Find f = new Find();
```



```

    }
    if(e.getSource()==zj){
        AddFI f = new AddFI();
    }
    if(e.getSource()==sc){
        Delet d = new Delet();
    }
    if(e.getSource()==xg){
        XiuGai x=new XiuGai();
    }
    if(e.getSource()==tc){
        shutDown();
    }
    if(e.getSource()==tj){
        Tongji t=new Tongji();
    }
}
private void shutDown()
{
    scs.stor();
    this.dispose();
}
}

```

信息已保存

/增加信息界面内部类，捕获文本框中信息创建 Student 对象，添加到 ArrayList 中，如果已存在该学号/给出提示信息，并重新添加。

```

class AddFI extends JFrame implements ActionListener {
    private JTextField STNOText, SNAMEText, MAText, CHIText, JAVAText;
    private JButton b1, b2, b3;
    private String STNO, SNAME,MAT, CHI, JAVA;
    public AddFI() {
        添加学生信息
        Container c2 = this.getContentPane();
        c2.setLayout(new GridLayout(3, 1));
        JPanel center = new JPanel(new GridLayout(5, 2));
        JPanel low = new JPanel(new FlowLayout());
        添加学生信息

        c2.add(label1);
        STNOText = new JTextField(30);//30 列文本框
        SNAMEText = new JTextField(30);
        CHIText = new JTextField(30);
        MAText = new JTextField(30);
    }
}

```

```
JAVAText = new JTextField(30);
    学号
center.add(STNOText);//添加文本框
    姓名
center.add(SNAMEText);
    语文
center.add(CHIText);
    数学
center.add(MAText);

center.add(JAVAText);
c2.add(center);
    添加
    清除
    退出

low.add(b1);
low.add(b2);
low.add(b3);
c2.add(low);
//添加监听
b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
this.setBounds(200, 200, 600, 400);
this.setVisible(true);
    添加学生信息
}

public void actionPerformed(ActionEvent e) {
    if (e.getSource() == b1) {
        try {
            addFI();
        } catch (FileNotFoundException e1) {
            e1.printStackTrace();
        } catch (IOException e1) {
            e1.printStackTrace();
        }
    }
    if (e.getSource() == b2) {
        clearForm();
    }
    if (e.getSource() == b3) {
        this.dispose();
    }
}
```

```

    }
    private void addFI() throws FileNotFoundException, IOException {
        STNO = STNOText.getText();
        SNAME = SNAMEText.getText();
        CHI = CHIText.getText();
        MAT = MATText.getText();
        JAVA = JAVAText.getText();
        if (STNO.length() == 0 || SNAME.length() == 0 || MAT.length() == 0
            || JAVA.length() == 0 || CHI.length() == 0)
            请添加完全信息
        else {
            Student a=new Student(SNAME,Integer.parseInt(STNO),
                Integer.parseInt(CHI),Integer.parseInt(MAT),Integer.parseInt(JAVA));
            int b=0;
            for(int i=0;i<scs.al.size();i++){
                if(scs.al.get(i).getNum()==Integer.parseInt(STNO)){
                    b=1;
                }
            }
            if(b==0){
                scs.adds(a);
                添加成功
            }
            else{
                已存在
            }
        }
    }
    private void clearForm() {

```

JAV

}/查询信息界面内部类,根据输入的学号,在 **arraylist** 中查找对应学号的学生信息,分别输出

```

class Find extends JFrame implements ActionListener {

    private JTextField STNOText, SNAMEText, MATText, CHIText, JAVAText;
    private String STNO;
    private JButton b1, b2;

```

```
public Find() {
    Container c1 = this.getContentPane();
    c1.setLayout(new GridLayout(4, 1));
        查询学生信息
        请输入你的学号

    JPanel pp = new JPanel(new GridLayout(2, 1));
    pp.add(label1);
    pp.add(label0);
    c1.add(pp);
    JPanel p1 = new JPanel();
    STNOText = new JTextField(10);
    p1.add(STNOText);
    c1.add(p1);
    JPanel p2 = new JPanel();
        查询
        退出

    b1.addActionListener(this);
    b2.addActionListener(this);
    p2.add(b1);
    p2.add(b2);
    c1.add(p2);

    JPanel center = new JPanel(new GridLayout(4, 2));
    SNAMEText = new JTextField(30);
    CHIText = new JTextField(30);
    MAText = new JTextField(30);
    JAVAText = new JTextField(30);
        姓名

    center.add(SNAMEText);
        语文

    center.add(CHIText);
        数学

    center.add(MAText);

    center.add(JAVAText);
    c1.add(center);
    this.setVisible(true);
    this.setBounds(200, 200, 400, 300);
}

public void actionPerformed(ActionEvent e){
    if (e.getSource() == b1) {
        STNO = STNOText.getText();
        int k=0;
        for(int i=0;i<scs.al.size();i++)
```

```

        {
            if(Integer.parseInt(STNO)==scs.al.get(i).getNum())
            {
                SNAMEText.setText(scs.al.get(i).getName());
                MAText.setText(String.valueOf(scs.al.get(i).getShuxue()));
                CHIText.setText(String.valueOf(scs.al.get(i).getYuwen()));
                JAVAText.setText(String.valueOf(scs.al.get(i).getJava()));
                k=1;
            }
        }
        if(k==0){
            查无此人
        }
    }
    if (e.getSource() == b2) {
        this.dispose();
    }
}
}

```

/删除信息界面，通过输入的学号进行查找并在 arraylist 中移除

```
class Delet extends JFrame implements ActionListener{
```

```
    private JButton yes;
```

```
    private JButton cangle;
```

```
    private JTextField text1;
```

```
    private String STNO;
```

```
    public Delet(){
```

```
        Container c3 = this.getContentPane();
```

```
        c3.setLayout(new GridLayout(3, 1));
```

```
        JPanel p1 = new JPanel();
```

```
        JPanel p2 = new JPanel();
```

删除学生信息

```
        label1.setForeground(Color.blue);
```

```
        c3.add(label1);
```

请输入学号

```
        text1 = new JTextField(10);
```

```
        p1.add(label2);
```

```
        p1.add(text1);
```

```
        c3.add(p1);
```

确定

退出

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/188021123013006035>