山东大学 软件 学院

面向对象 课程实验报告

|  |  |
| --- | --- |
| 实验题目：猜数字游戏 | |
| 实验学时：16 | 实验日期： 2023.5.8 |
| 实验目的：  《面向对象技术》是计算机科学与技术、软件工程专业学生的一门必修课程。面向对象技术是当前最重要的程序设计思想与方法，利用抽象性、封装性、继承性、多态性等特性帮助人们开发出模块化、复用性强、易修改、可扩展的软件系统和程序。本实验通过若干个备选的相对复杂、独立的题目，使学生在实验过程中全面、系统的熟悉面向对象技术的基本概念和设计技术，分析、设计一个相对完整的应用，并使用面向对象编程语言进行实现，通过实验掌握使用面向对象技术进行程序设计、开发的思想和技能 | |
| 硬件环境：  CPU 2GHz、内存4GB、存储空间5GB。 | |
| 软件环境：  IntelliJ IDEA、OpenJDK-17 | |
| 实验步骤与内容：  1.实验要求：  完成一个猜数字的GUI游戏。  开始游戏后，产生一个没有重复数字的4位随机数，用户每猜一个数字，显示出“完全猜中的数字个数”和“猜中数字但位置错误的数字个数”，比如nAmB，数字n表示猜中的位置正确的数字个数，数字m表示数字正确而位置不对的数字个数。例如，正确答案为5234，如果用户猜5346，则显示：1A2B，数字1表示数字5及其位置猜对了，数字3和4这两个数字猜对了，但是位置没对，记为2B。然后，用户根据游戏提示的信息继续猜，直到猜中为止。同时设计规则，根据猜中的次数计算积分，并可以显示不同用户的排行榜。   1. **实验步骤：**   （1）写出核心猜数字算法。  （2）建立对象类如用户类等，添加属性方法等。  （3）将控制台输出文字逐渐修改为由图形界面展示。  （4）搭建GUI界面，创建各种面板及按钮等，包括开始面板、游戏界面、排行界面和选项列表。  （5）规定各个按钮的响应事件。  （6）进行测试，调整代码顺序以使游戏正常进行。  （7）生成可执行文件。  （8）完成实验报告及设计文档。   1. **实现界面：**   如下图所示：  （1）  **b4d3326207dc32a7ec6258f62efe732**  （2）  **7826d324e91964d99dfef1ab947def6**  （3）  **65f742f66b6f2064a34cffded17ff6e** | |
| 结论分析与体会：  这次面向对象实验，我选择的是猜数字题目，题目比较简单，猜数字的核心思想也比较简单。我主要是在GUI界面上花的功夫比较多，这是第一次用eclipse来画界面。从这次实验中，我了解到了很多java里面的控件，并且初步地去尝试了怎么使用他们，对于界面布局、控件使用、监听器、容器等等也有了更深入的理解。并且通过这次实验，加深了在课堂上学习的继承、实现、关联、依赖等面向对象知识，对于大一所学的java又有了更加深入的了解。在这次实验中，我也初步尝试画UML图，可能画得还不够完善，但我也从中切切实实地感受到了用UML图，能够更加简明地表达自己的代码思想和算法。总而言之，在这次实验中，我学到了很多，也希望自己接下去的学习也能够收获颇丰！ | |

**实验代码：**

**import com.sun.istack.internal.NotNull;**

**import java.awt.FlowLayout;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import java.awt.Font;**

**import javax.swing.\*;**

**import java.awt.Font;**

**import javax.swing.JButton;**

**import javax.swing.JFrame;**

**public class ActMenu extends JPanel{**

**JButton buttons[] = {new JButton("新用户"),new JButton("猜新数"),new JButton("排名")};**

**public ActMenu() {**

**Font font=new Font("楷体",Font.BOLD,20);//设置字体**

**buttons[0].setFont(font);//设置按钮字体**

**buttons[1].setFont(font);//设置按钮字体**

**buttons[2].setFont(font);//设置按钮字体**

**setLayout(new FlowLayout());**

**for (int index = 0; index < buttons.length; index++)**

**buttons[index].addActionListener(new ActLis());**

**for (int index = 0; index < buttons.length; index++)**

**add(buttons[index]);**

**}**

**public void shutdown() {**

**for (int index = 0; index < run.frames.length; index++)**

**run.frames[index].setVisible(false);**

**}**

**private class ActLis implements ActionListener {**

**public void actionPerformed(@NotNull ActionEvent e) {**

**if (e.getSource() == buttons[0]) {**

**shutdown();**

**if(Game.count>=1&&Game.users[Game.count - 2].name == null) Game.count--;**

**run.frames[1].setVisible(true);**

**new Login();**

**Game.temp=-1;**

**//Game.count++;**

**//Game.mem\_count=Game.count;**

**run.game.greet.setText(Client.greetings(Game.count));**

**resetuser();**

**//Game.users[Game.count-1]=new Client();**

**run.game.process();**

**//System.out.println(Game.count);**

**}**

**else if (e.getSource() == buttons[1]) {**

**shutdown();**

**if(Game.count>=1&&Game.users[Game.count - 2].name == null) Game.count--;**

**run.frames[1].setVisible(true);**

**resetnum();**

**run.game.process();**

**}**

**else if (e.getSource() == buttons[2]) {**

**shutdown();**

**//System.out.println(Game.count-1);**

**if(Game.count>=1&&Game.users[Game.count - 2].name == null) Game.count--;**

**run.frames[2].setVisible(true);**

**run.rank.d.setText(run.rank.printRank());**

**}**

**}**

**}**

**/\*ActMenu.addWindowListener(new WindowAdapter()**

**{**

**@Override**

**public void windowClosing(WindowEvent e)**

**{**

**System.out.println("Closed");**

**e.getWindow().dispose();**

**}**

**});\*/**

**public void resetnum(){**

**run.game.guess.setText("");**

**run.game.scores.setText("");**

**run.game.note.setText("");**

**run.game.str="";**

**}**

**public void resetuser(){**

**run.game.nam.setText("");**

**run.game.guess.setText("");**

**run.game.scores.setText("");**

**run.game.note.setText("");**

**run.game.str="";**

**}**

**}**

**import java.io.\*;**

**import java.util.regex.Matcher;**

**import java.util.regex.Pattern;**

**public class Check {**

**/\*\***

**\* 在登录时， 验证账号密码是否正确**

**\*/**

**public boolean check1(String countname,String pwd) throws IOException{**

**File file=new File("Message.txt"); //创建文件类**

**if(!file.exists()||file.isDirectory()) //判断文件是否存在**

**//throw new FileNotFoundException();**

**file.createNewFile();**

**//BufferedReader br=new BufferedReader(new FileReader(file)); //创建读入缓冲流，按行读入**

**InputStreamReader fReader = new InputStreamReader(new FileInputStream("Message.txt"),"GB2312");**

**BufferedReader br = new BufferedReader(fReader);**

**//BufferedReader br = new BufferedReader( new InputStreamReader(file,"GB2312"));**

**String temp=null;**

**// StringBuffer sb=new StringBuffer();**

**temp=br.readLine(); //先读取一行**

**while(temp!=null){**

**String sbstring = temp.toString(); //转化为string**

**int n = sbstring.length(); //测字符串长度**

**String []message = new String[5]; //按~拆分 成5个字符串数组，按账号和密码进行信息验证**

**int k=0;**

**for (int i=0; i<2; i++)**

**message[i]="";**

**//我们在写入用户时用~分割， 所以我们利用~在分割开来**

**for (int i=0; i<n; i++)**

**{**

**if(sbstring.charAt(i)=='~')**

**{**

**//System.out.println("@"+message[k]);**

**k++;**

**}**

**else**

**{**

**message[k] += sbstring.charAt(i);**

**}**

**}**

**//System.out.println(countname);**

**//System.out.println(message[0]);**

**if (countname.equals(message[0])&&pwd.equals(message[1]))//比较账户密码是否相等**

**return true;**

**temp=br.readLine(); //读取下一行**

**}**

**return false;**

**}**

**//在注册时 验证账号是否存在**

**public boolean check2(String countname) throws IOException{**

**File file=new File("Message.txt"); //创建文件类**

**if(!file.exists()||file.isDirectory()) //判断文件是否存在**

**file.createNewFile();**

**BufferedReader br=new BufferedReader(new FileReader(file)); //创建读入缓冲流，按行读入**

**String temp=null;**

**// StringBuffer sb=new StringBuffer();**

**temp=br.readLine(); //先读取一行**

**while(temp!=null){**

**String sbstring = temp.toString(); //转化为string**

**int n = sbstring.length(); //测字符串长度**

**String []message = new String[5]; //按~拆分 成5个字符串数组，按账号和密码进行信息验证**

**int k=0;**

**for (int i=0; i<5; i++)**

**message[i]="";**

**for (int i=0; i<n; i++)**

**{**

**if(sbstring.charAt(i)=='~')**

**{**

**//System.out.println("@"+message[k]);**

**k++;**

**}**

**else**

**{**

**message[k] += sbstring.charAt(i);**

**}**

**}**

**if (countname.equals(message[0]))**

**return true;**

**temp=br.readLine();**

**}**

**return false;**

**}**

**//验证用户名和密码是否为中文**

**/\*public boolean checkcountname(String countname)**

**{**

**Pattern p = Pattern.compile("[\u4e00-\u9fa5]");**

**Matcher m = p.matcher(countname);**

**if (m.find()) {**

**return true;**

**}**

**return false;**

**}**

**//验证姓名是否为中文**

**public boolean checkname(String name)**

**{**

**int n = 0;**

**for(int i = 0; i < name.length(); i++) {**

**n = (int)name.charAt(i);**

**if(!(19968 <= n && n <40869)) {**

**return false;**

**}**

**}**

**return true;**

**}\*/**

**}**

**import java.util.\*;**

**public class Client {**

**String name;**

**int scores;**

**public void winscores(){**

**scores++;**

**}**

**public static String greetings(int num){**

**String str=""+"当前用户:";**

**return str;**

**}**

**}**

**import java.awt.\*;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import java.awt.event.WindowAdapter;**

**import java.awt.event.WindowEvent;**

**import java.io.File;**

**import java.io.FileOutputStream;**

**import java.io.IOException;**

**import java.io.\*;**

**import javax.swing.\*;**

**public class Game extends JFrame{**

**static int length = 4;**

**static int count;**

**static int temp=-1;**

**static int mem\_count=count;**

**static Client[] users=new Client[100];**

**JLabel center = new JLabel();**

**JLabel first=new JLabel();**

**JTextField greet = new JTextField();**

**static JTextField nam = new JTextField();**

**JLabel second=new JLabel();**

**JTextField correct=new JTextField();**

**JTextField start = new JTextField();**

**JLabel third = new JLabel();**

**JTextField guess=new JTextField();**

**JButton startguess = new JButton("确定");**

**JLabel forth = new JLabel();**

**JTextField scores = new JTextField();**

**JTextArea note = new JTextArea();**

**JScrollPane sc=new JScrollPane(note);**

**Guess one = new Guess(length);**

**int[] rnum = new int[length];**

**String str="";**

**public Game(){**

**super("猜数字");**

**Font font=new Font("楷体",Font.BOLD,30);//设置字体**

**startguess.setFont(font);//设置按钮字体**

**setLayout(new BorderLayout());**

**add(new ActMenu(), BorderLayout.NORTH);**

**add(center,BorderLayout.CENTER);**

**center.setLayout(new GridLayout(4,1));**

**greet.setFont(new Font("楷体",Font.BOLD,25) );**

**greet.setHorizontalAlignment(JTextField.CENTER);**

**nam.setFont(new Font("楷体",Font.PLAIN,30) );**

**nam.setHorizontalAlignment(JTextField.CENTER);**

**center.add(first);**

**first.setLayout(new GridLayout(1,2));**

**first.add(greet);**

**nam.setEditable(false);**

**first.add(nam);**

**greet.setEditable(false);**

**greet.setText(Client.greetings(count));**

**center.add(second);**

**second.setLayout(new GridLayout(1,1));**

**//second.add(correct);**

**second.add(start);**

**start.setHorizontalAlignment(JTextField.CENTER);**

**correct.setEditable(false);**

**start.setEditable(false);**

**start.setFont(new Font("楷体",Font.PLAIN,50) );**

**start.setText("请输入数字进行游戏");**

**guess.setFont(new Font("楷体",Font.PLAIN,30) );**

**guess.setHorizontalAlignment(JTextField.CENTER);**

**center.add(third);**

**third.setLayout(new GridLayout(1,2));**

**third.add(guess);**

**third.add(startguess);**

**startguess.addActionListener(new ActLis());**

**scores.setFont(new Font("楷体",Font.PLAIN,30) );**

**scores.setHorizontalAlignment(JTextField.CENTER);**

**note.setFont(new Font("楷体",Font.PLAIN,18));**

**center.add(forth);**

**forth.setLayout(new GridLayout(1,2));**

**forth.add(scores);**

**forth.add(sc);**

**scores.setEditable(false);**

**note.setOpaque(false);**

**//users[count-1]=new Client();**

**process();**

**}**

**public void process() {**

**rnum= one.rand(rnum);**

**correct.setText(one.correct(rnum));**

**}**

**private class ActLis implements ActionListener {**

**public void actionPerformed(ActionEvent e) {**

**if (e.getSource() == startguess) {**

**int[] gnum = new int[length];**

**String gstr=guess.getText();**

**gnum=one.guess(gstr,gnum);**

**for(int i=0;i<=count-1;i++){**

**if(users[i].name==null) continue;**

**if(users[i].name.equals(nam.getText())){**

**temp=i;**

**break;**

**}**

**}**

**//System.out.println(nam.getText().equals(""));**

**if(nam.getText().equals("")==true){**

**JOptionPane.showMessageDialog(null, "请您登录后再进行游戏", "尚未登录！", JOptionPane.ERROR\_MESSAGE);**

**}**

**if(temp==-1&&nam.getText().equals("")==false) {**

**Game.count++;**

**Game.users[Game.count-1]=new Client();**

**str += one.compare(rnum, gnum, users[count - 1]);**

**note.setText(str);**

**users[count - 1].name = nam.getText();**

**scores.setText("" + users[count - 1].name + "当前得分:" + users[count - 1].scores);**

**}**

**else if(temp!=-1&&nam.getText().equals("")==false){**

**//if(mem\_count==count) count--;**

**str += one.compare(rnum, gnum, users[temp]);**

**note.setText(str);**

**scores.setText("" + users[temp].name + "当前得分:" + users[temp].scores);**

**}**

**run.rank.printRank();**

**}**

**}**

**}**

**}**

**import java.util.Random;**

**import java.util.Scanner;**

**import java.util.\*;**

**public class Guess {**

**int length;**

**public Guess(int lenth){**

**this.length = lenth;**

**}**

**public int[] rand(int[] rnum) {**

**Random random = new Random();**

**for (int i = 0; i < rnum.length;) {**

**int number = random.nextInt(10);**

**boolean uncopy = true;**

**for (int j = 0; j < i; j++) {**

**if (number == rnum[j])**

**uncopy = false;**

**}**

**if (uncopy) {**

**rnum[i] = number;**

**i++;**

**}**

**}**

**return rnum;**

**}**

**public int[] guess(String str, int[] gnum){**

**char[] cs = str.toCharArray();**

**for(int i=0; i<gnum.length; i++)**

**gnum[i]=cs[i]-'0';**

**return gnum;**

**}**

**public String compare(int[] rnum,int[] gnum,Client user){**

**int n=0,m=0,len=rnum.length;**

**boolean[] exact = new boolean[len];**

**for(int i=0;i<len;i++)**

**if(rnum[i]==gnum[i]){**

**n++;**

**exact[i]=true;**

**}**

**for(int i=0;i<len;i++){**

**if(exact[i])**

**continue;**

**for(int j=0;j<len;j++)**

**if(gnum[i]==rnum[j])**

**m++;**

**}**

**String result;**

**if(n==len){**

**result = n+"A"+" 猜对了!\n";**

**user.winscores();**

**}**

**else{**

**result = n+"A"+m+"B"+" 再试试吧!\n";**

**}**

**return result;**

**}**

**public String correct(int[] rnum){**

**String str="正确答案为：";**

**for(int i=0;i<rnum.length;i++)**

**str+=rnum[i];**

**System.out.println(str);**

**return str;**

**}**

**}**

**import java.awt.\*;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import java.io.IOException;**

**import javax.swing.\*;**

**@SuppressWarnings("serial")**

**public class Login extends JFrame implements ActionListener{**

**JButton jb1, jb2, jb3; //按钮**

**JPanel jp1,jp2,jp3, jp4; //面板**

**JTextField jtf; //文本框**

**JLabel jlb1, jlb2, jlb3; //标签**

**JPasswordField jpf; //密码框**

**static String Name;**

**public Login() {**

**// TODO Auto-generated constructor stub**

**jb1 = new JButton("登录");**

**jb1.setFont(new Font("楷体",Font.BOLD,18) );**

**jb2 = new JButton("注册");**

**jb2.setFont(new Font("楷体",Font.BOLD,18) );**

**//设置按钮监听**

**jb1.addActionListener(this);**

**jb2.addActionListener(this);**

**jp1 = new JPanel(); //创建面板**

**jp2 = new JPanel();**

**jp3 = new JPanel();**

**jp4 = new JPanel();**

**jlb1 = new JLabel("用户名"); //添加标签**

**jlb1.setFont(new Font("楷体",Font.BOLD,16) );**

**jlb2 = new JLabel("密 码");**

**jlb2.setFont(new Font("楷体",Font.BOLD,16) );**

**jtf = new JTextField(10); //创建文本框和密码框**

**jtf.setFont(new Font("serial",Font.PLAIN,16) );**

**jpf = new JPasswordField(10);**

**jpf.setFont(new Font("serial",Font.PLAIN,16) );**

**//加入面板中**

**jp1.add(jlb1);**

**jp1.add(jtf);**

**jp2.add(jlb2);**

**jp2.add(jpf);**

**jp3.add(jb1);**

**jp3.add(jb2);**

**//将JPane加入JFrame中**

**this.add(jp1);**

**this.add(jp2);**

**this.add(jp3);**

**//设置布局**

**this.setTitle("用户登录");**

**this.setLayout(new GridLayout(3,1));**

**this.setSize(300, 200); //设置窗体大小**

**this.setLocationRelativeTo(null);//在屏幕中间显示(居中显示)**

**this.setDefaultCloseOperation(DISPOSE\_ON\_CLOSE); //设置仅关闭当前窗口**

**this.setVisible(true); //设置可见**

**this.setResizable(false); //设置不可拉伸大小**

**}**

**@Override**

**public void actionPerformed(ActionEvent e) {**

**// TODO Auto-generated method stub**

**if (e.getActionCommand()=="登录")**

**{**

**try {**

**Name=jtf.getText();**

**Game.nam.setText(Name);**

**login();**

**} catch (HeadlessException | IOException e1) {**

**// TODO Auto-generated catch block**

**e1.printStackTrace();**

**}**

**}**

**else if(e.getActionCommand()=="注册")**

**{**

**// clear();**

**new Register();**

**dispose(); //使窗口消失**

**}**

**}**

**//清空账号和密码框**

**private void clear() {**

**// TODO Auto-generated method stub**

**jtf.setText(""); //设置为空**

**jpf.setText("");**

**}**

**//验证登录信息，并做处理**

**@SuppressWarnings("deprecation")**

**public void login() throws HeadlessException, IOException**

**{**

**//判断账户名和密码是否包含中文**

**/\*if (new Check().checkcountname(jtf.getText())||new Check().checkcountname(jpf.getText()))**

**{**

**JOptionPane.showMessageDialog(null, "用户名或密码存在中文，不合法!","消息提示",JOptionPane.WARNING\_MESSAGE);**

**Game.nam.setText(null);**

**}\*/**

**if(jtf.getText().isEmpty()&&jpf.getText().isEmpty())**

**{**

**JOptionPane.showMessageDialog(null, "账号密码为空，请输入！","消息提示",JOptionPane.WARNING\_MESSAGE);**

**Game.nam.setText(null);**

**}**

**else if (jtf.getText().isEmpty())**

**{**

**JOptionPane.showMessageDialog(null, "账号为空，请输入！","消息提示",JOptionPane.WARNING\_MESSAGE);**

**Game.nam.setText(null);**

**}**

**else if (jpf.getText().isEmpty())**

**{**

**JOptionPane.showMessageDialog(null, "密码为空，请输入！","消息提示",JOptionPane.WARNING\_MESSAGE);**

**Game.nam.setText(null);**

**}**

**else if (new Check().check1(jtf.getText(),jpf.getText()))**

**{**

**JOptionPane.showMessageDialog(null,"登录成功！","提示消息",JOptionPane.WARNING\_MESSAGE);**

**dispose(); //使文原窗体消失**

**}**

**else**

**{**

**JOptionPane.showMessageDialog(null, "账号密码错误请重新输入！","消息提示",JOptionPane.ERROR\_MESSAGE);**

**Game.nam.setText(null);**

**clear(); //调用清除函数**

**}**

**}**

**}**

**import java.awt.BorderLayout;**

**import java.awt.Font;**

**import java.awt.GridLayout;**

**import java.awt.event.WindowAdapter;**

**import java.awt.event.WindowEvent;**

**import java.io.File;**

**import java.io.FileNotFoundException;**

**import java.io.FileOutputStream;**

**import java.io.IOException;**

**import javax.swing.\*;**

**public class Rank extends JFrame{**

**JTextArea d = new JTextArea();**

**public Rank(){**

**super("猜数字");**

**setLayout(new BorderLayout());**

**add(new ActMenu(), BorderLayout.NORTH);**

**JScrollPane scrollPane = new JScrollPane(**

**d,**

**ScrollPaneConstants.VERTICAL\_SCROLLBAR\_ALWAYS,**

**ScrollPaneConstants.HORIZONTAL\_SCROLLBAR\_NEVER**

**);**

**//scrollPane.setContentPane(contentPane);**

**add(scrollPane,BorderLayout.CENTER);**

**d.setEditable(false);**

**d.setOpaque(false);**

**d.setLineWrap(true);**

**d.setFont(new Font("楷体",0,22));**

**this.addWindowListener(new WindowAdapter()**

**{**

**@Override**

**public void windowClosing(WindowEvent e)**

**{**

**System.out.println("Closed");**

**e.getWindow().dispose();**

**}**

**});**

**}**

**public Client[] rank(){**

**Client[] cli=new Client[Game.count];**

**for(int i=0;i<cli.length;i++)**

**cli[i]=Game.users[i];**

**for(int i=0;i<Game.count;i++){**

**Client temp=cli[i];**

**int n=i;**

**for(int j=i+1;j<Game.count;j++){**

**//System.out.println(j);**

**//System.out.println(cli[j].scores);**

**if(cli[j].scores>temp.scores){**

**temp=cli[j];**

**n=j;**

**}**

**}**

**cli[n]=cli[i];**

**cli[i]=temp;**

**}**

**return cli;**

**}**

**public String printRank(){**

**Client[] cli=rank();**

**String str="\t 排 名 用 户 名 总 分\n\n";**

**String temp\_name,temp\_score,temp\_no;**

**for(int i=0;i<Game.count;i++) {**

**temp\_name=cli[i].name;**

**int temp\_len=cli[i].name.length();**

**for(int j=0;j<cli[i].name.length();j++){**

**if((cli[i].name.charAt(j) >= 0x4e00)&&(cli[i].name.charAt(j) <= 0x9fbb)) {temp\_len++;}**

**}**

**for(int j=1;j<=18-temp\_len;j++)**

**temp\_name+=" ";**

**temp\_score = String.valueOf(cli[i].scores);**

**for(int j=1;j<=12-temp\_score.length();j++)**

**temp\_score+=" ";**

**temp\_no = String.valueOf(i+1);**

**for(int j=1;j<=16-temp\_no.length();j++)**

**temp\_no+=" ";**

**str += "\t " +temp\_no+ " " + temp\_name + temp\_score + "\n\n";**

**}**

**File file = new File("Rank.txt");**

**String messagesum = "";**

**for (int i = 0; i < Game.count; i++) {**

**if(cli[i].name==null) continue;**

**messagesum += cli[i].name + "~";**

**messagesum+=String.valueOf(cli[i].scores);**

**if(i!=Game.count-1) messagesum+="\n";**

**}**

**try {**

**file.createNewFile();**

**} catch (IOException e) {**

**e.printStackTrace();**

**}**

**FileOutputStream out = null; //建立输出对象，true表示追加**

**try {**

**out = new FileOutputStream(file, false);**

**} catch (FileNotFoundException fileNotFoundException) {**

**fileNotFoundException.printStackTrace();**

**}**

**StringBuffer sb = new StringBuffer(); //创建字符串流**

**sb.append(messagesum + "\n"); //向字符串流中添加信息**

**try {**

**out.write(sb.toString().getBytes("gb2312")); //将字符串流中的信息写入文本**

**} catch (IOException ioException) {**

**ioException.printStackTrace();**

**}**

**try {**

**out.close(); //关闭**

**} catch (IOException ioException) {**

**ioException.printStackTrace();**

**}**

**return str;**

**}**

**}**

**import java.awt.\*;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import java.io.IOException;**

**import javax.swing.\*;**

**/\*\***

**\* 此类完成对注册页面的编写， 用户需填写 账户，密码**

**\***

**\* 并且会进行验证操作， 如姓名是否合法（中文）**

**\***

**\*/**

**@SuppressWarnings("serial")**

**public class Register extends JFrame implements ActionListener{**

**JButton jb1, jb2; //按钮**

**JLabel jlb1, jlb2, jlb3,jlb4,jlb5, jlb6; //标签**

**JTextField jtf3,jtf4; //文本框**

**JPasswordField jpf; //密码框**

**JPanel jp3, jp4,jp6,jp7; //面板**

**public Register() {**

**// TODO Auto-generated constructor stub**

**//按钮**

**jb1 = new JButton("提交");**

**jb1.setFont(new Font("楷体",Font.BOLD,18) );**

**jb2 = new JButton("登录");**

**jb2.setFont(new Font("楷体",Font.BOLD,18) );**

**//设置按钮监听**

**jb1.addActionListener(this);**

**jb2.addActionListener(this);**

**//标签信息**

**jlb3 = new JLabel(" 账号");**

**jlb3.setFont(new Font("楷体",Font.BOLD,16) );**

**jlb4 = new JLabel(" 密码");**

**jlb4.setFont(new Font("楷体",Font.BOLD,16) );**

**jtf3 = new JTextField(10);**

**jtf3.setFont(new Font("serial",Font.PLAIN,16) );**

**jtf4 = new JTextField(10);**

**jtf4.setFont(new Font("serial",Font.PLAIN,16) );**

**jp3 = new JPanel();**

**jp4 = new JPanel();**

**jp6 = new JPanel();**

**jp7 = new JPanel();**

**//将对应信息加入面板中**

**jp3.add(jlb3);**

**jp3.add(jtf3);**

**jp4.add(jlb4);**

**jp4.add(jtf4);**

**jp6.add(jb1);**

**jp6.add(jb2);**

**//将JPane加入JFrame中**

**//this.add(jp7); //先加入提示语**

**this.add(jp3);**

**this.add(jp4);**

**this.add(jp6);**

**//设置布局**

**this.setTitle("注册信息");**

**this.setLayout(new GridLayout(3,1));**

**this.setSize(300, 200); //设置窗体大小**

**this.setLocationRelativeTo(null);//在屏幕中间显示(居中显示)**

**this.setDefaultCloseOperation(DISPOSE\_ON\_CLOSE); //设置仅关闭当前窗口**

**this.setVisible(true); //设置可见**

**this.setResizable(false); //设置不可拉伸大小**

**}**

**@Override**

**public void actionPerformed(ActionEvent e) {**

**// TODO Auto-generated method stub**

**if (e.getActionCommand()=="提交")**

**{**

**try {**

**register();**

**} catch (IOException e1) {**

**// TODO Auto-generated catch block**

**e1.printStackTrace();**

**}**

**}**

**else if (e.getActionCommand()=="登录")**

**{**

**new Login();**

**}**

**}**

**//验证注册信息，并做处理**

**public void register() throws IOException**

**{**

**//判断信息是否补全**

**if (jtf3.getText().isEmpty()||jtf4.getText().isEmpty())**

**{**

**JOptionPane.showMessageDialog(null, "信息有空缺，请补全！","消息提示",JOptionPane.WARNING\_MESSAGE);**

**}**

**//判断账户名和密码是否包含中文**

**/\*else if (new Check().checkcountname(jtf3.getText())||new Check().checkcountname(jtf4.getText()))**

**{**

**JOptionPane.showMessageDialog(null, "用户名或密码存在中文，不合法!","消息提示",JOptionPane.WARNING\_MESSAGE);**

**}\*/**

**//满足要求**

**else if (!jtf3.getText().isEmpty()&&!jtf4.getText().isEmpty())**

**{**

**//注册成功， 打包为信息数组传递给UserMessage进行更新操作**

**String []message = new String[2];**

**message[0] = jtf3.getText();**

**message[1] = jtf4.getText();**

**if (!new Check().check2(message[0])) //调用Check的check方法检测用户是否存在， 如果不存在执行**

**{**

**new UserMessage().write(message); //调用UserMseeage的write方法进行写操作， 将信息格式化存入**

**JOptionPane.showMessageDialog(null,"注册成功！","提示消息",JOptionPane.WARNING\_MESSAGE);**

**dispose(); //使窗口消失**

**new Login();**

**}**

**else**

**{**

**JOptionPane.showMessageDialog(null,"账号已存在，请重新输入！","提示消息",JOptionPane.WARNING\_MESSAGE);**

**//dispose();**

**}**

**}**

**}**

**//清空账号和密码框**

**private void clear() {**

**// TODO Auto-generated method stub**

**jtf3.setText("");**

**jtf4.setText("");**

**}**

**}**

**import java.awt.Dimension;**

**import java.awt.Toolkit;**

**import java.io.\*;**

**import javax.swing.JFrame;**

**public class run {**

**static Start start=new Start();**

**static Game game=new Game();**

**static Rank rank=new Rank();**

**static JFrame[] frames = { start,game,rank};**

**public static void main(String args[]) throws InterruptedException {**

**File file=new File("Rank.txt"); //创建文件类**

**Game.count=0;**

**try {**

**file.createNewFile();**

**} catch (IOException e) {**

**e.printStackTrace();**

**}**

**String temp=null;**

**try {**

**//BufferedReader br=new BufferedReader(new FileReader(file)); //创建读入缓冲流，按行读入**

**InputStreamReader fReader = new InputStreamReader(new FileInputStream("Rank.txt"),"GB2312");**

**BufferedReader br = new BufferedReader(fReader);**

**temp=br.readLine(); //先读取一行**

**while(temp!=null){**

**String sbstring = temp.toString(); //转化为string**

**int n = sbstring.length(); //测字符串长度**

**String []message = new String[5]; //按~拆分 成5个字符串数组，按账号和密码进行信息验证**

**int k=0;**

**for (int i=0; i<2; i++)**

**message[i]="";**

**//我们在写入用户时用~分割， 所以我们利用~在分割开来**

**for (int i=0; i<n; i++)**

**{**

**if(sbstring.charAt(i)=='~')**

**{**

**//System.out.println("@"+message[k]);**

**k++;**

**}**

**else**

**{**

**message[k] += sbstring.charAt(i);**

**}**

**}**

**Game.temp=-1;**

**Game.count++;**

**Game.mem\_count=Game.count;**

**Game.users[Game.count-1]=new Client();**

**Game.users[Game.count-1].name=message[0];**

**if(Game.users[Game.count-1].name==null||Game.users[Game.count-1].name.equals("")) {Game.count--;**

**System.out.println("!!!");**

**break;**

**}**

**//System.out.println(Game.users[Game.count-1].name);**

**Game.users[Game.count-1].scores=Integer.parseInt(message[1]);**

**temp=br.readLine(); //读取下一行**

**}**

**} catch (FileNotFoundException fileNotFoundException) {**

**fileNotFoundException.printStackTrace();**

**} catch (IOException e) {**

**e.printStackTrace();**

**}**

**Game.temp=-1;**

**//Game.count++;**

**//System.out.println(Game.count-1);**

**for (int index = 0; index < frames.length; index++) {**

**frames[index].setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);**

**frames[index].setPreferredSize(new Dimension(700,750));**

**screencentre(frames[index]);**

**frames[index].pack();**

**}**

**frames[0].setVisible(true);**

**// StringBuffer sb=new StringBuffer();**

**}**

**//使框架显示在屏幕中央**

**public static void screencentre(JFrame frame) {**

**Toolkit kit = Toolkit.getDefaultToolkit(); // 定义工具包**

**Dimension screenSize = kit.getScreenSize(); // 获取屏幕的尺寸**

**int screenWidth = screenSize.width; // 获取屏幕的宽**

**int screenHeight = screenSize.height; // 获取屏幕的高**

**frame.setLocation(screenWidth/4, screenHeight/15);// 设置窗口居中显示**

**}**

**}**

**import java.awt.\*;**

**import java.awt.event.ActionEvent;**

**import java.awt.event.ActionListener;**

**import javax.swing.\*;**

**public class Start extends JFrame{**

**JPanel panel =new JPanel();**

**JLabel title = new JLabel("猜数字",JLabel.CENTER);**

**JButton[] buttons = {**

**new JButton("新的游戏"),**

**new JButton ("查看排行"),**

**new JButton("退出") };**

**JPanel[] panels = new JPanel[3];**

**public Start() {**

**super("猜数字");**

**Font font=new Font("楷体",Font.BOLD,25);//设置字体**

**buttons[0].setFont(font);//设置按钮字体**

**buttons[1].setFont(font);//设置按钮字体**

**buttons[2].setFont(font);//设置按钮字体**

**setLayout(new GridLayout(5,1));**

**title.setFont(new Font("楷体",Font.BOLD,100) );**

**add(title);**

**add(new JPanel());**

**for(int index=0;index<buttons.length;index++){**

**buttons[index].addActionListener(new StartLis());**

**panels[index]=new JPanel();**

**panels[index].add(buttons[index]);**

**add(panels[index]);**

**}**

**}**

**private class StartLis implements ActionListener {**

**public void actionPerformed(ActionEvent e) {**

**if(e.getSource()==buttons[0]){**

**setVisible(false);**

**run.rank.d.setText(run.rank.printRank());**

**//Game.count++;**

**run.frames[1].setVisible(true);**

**new Login();**

**//System.out.println(Game.count);**

**}**

**else if(e.getSource()==buttons[1]){**

**setVisible(false);**

**run.frames[2].setVisible(true);**

**run.rank.d.setText(run.rank.printRank());**

**}**

**else if(e.getSource()==buttons[2]){**

**System.exit(0);**

**}**

**}**

**}**

**}**

**import java.io.\*;**

**/\*\***

**\* 此类事对用户信息的 写入和读取操作**

**\***

**\*/**

**public class UserMessage**

**{**

**/\***

**\* 将注册的信息写入文本**

**\*/**

**public void write(String[] message)throws IOException**

**{**

**File file=new File("Message.txt");**

**String messagesum="";**

**for (int i=0; i<2; i++) //将信息格式化存储**

**messagesum+=message[i]+"~";**

**if(!file.exists())**

**file.createNewFile();**

**FileOutputStream out=new FileOutputStream(file,true); //建立输出对象，true表示追加**

**StringBuffer sb=new StringBuffer(); //创建字符串流**

**sb.append(messagesum+"\n"); //向字符串流中添加信息**

**out.write(sb.toString().getBytes("gb2312")); //将字符串流中的信息写入文本**

**out.close(); //关闭**

**}**

**/\***

**\* 读取信息，将用户名信息返回（如果不存在返回null），和Check类配合使用**

**\*/**

**public String[] read(String countname) throws IOException**

**{**

**File file=new File("Message.txt");**

**if(!file.exists()||file.isDirectory())**

**throw new FileNotFoundException();**

**BufferedReader br=new BufferedReader(new FileReader(file));**

**String temp=null;**

**StringBuffer sb=new StringBuffer();**

**temp=br.readLine();**

**String []message = new String[5]; //按~拆分 成5个字符串数组，按账号和密码进行信息验证**

**while(temp!=null){**

**String sbstring = temp.toString();**

**int n = sbstring.length(); //测字符串长度**

**for (int i=0; i<5; i++)**

**message[i] = "";**

**int k=0;**

**for (int i=0; i<n; i++)**

**{**

**if(sbstring.charAt(i)=='~')**

**{**

**//System.out.println("@"+message[k]);**

**k++;**

**}**

**else**

**{**

**message[k] += sbstring.charAt(i);**

**}**

**}**

**if (message[2].equals(countname)) //返回找到用户的信息**

**{**

**return message;**

**}**

**temp=br.readLine();**

**}**

**return null;**

**}**

**//**

**//更新密码**

**public String updatepwd(String countname,String pwd) throws IOException**

**{**

**File file=new File("Message.txt");**

**if(!file.exists()||file.isDirectory())**

**throw new FileNotFoundException();**

**//读文件**

**BufferedReader br=new BufferedReader(new FileReader(file));**

**String temp=null;**

**StringBuffer sb=new StringBuffer();**

**//写文件**

**//FileOutputStream out=new FileOutputStream(file,false);**

**StringBuffer sb1=new StringBuffer();**

**String moneystring="";**

**temp=br.readLine();**

**String []message = new String[2]; //按~拆分 成5个字符串数组，按账号和密码进行信息验证**

**while(temp!=null){**

**String sbstring = temp.toString();**

**int n = sbstring.length(); //测字符串长度**

**for (int i=0; i<2; i++)**

**message[i] = "";**

**int k=0;**

**for (int i=0; i<n; i++) //拆乘5个String**

**{**

**if(sbstring.charAt(i)=='~')**

**{**

**//System.out.println("@"+message[k]);**

**k++;**

**}**

**else**

**{**

**message[k] += sbstring.charAt(i);**

**}**

**}**

**if (message[0].equals(countname)) //找到该账户名**

**{**

**//修改密码**

**for (int i=0; i<3; i++)**

**sb1.append(message[i]+"~");**

**sb1.append(pwd+"~");**

**sb1.append(message[4]+"~\n");**

**}**

**else**

**{**

**sb1.append(temp+"\n");**

**}**

**temp=br.readLine();**

**}**

**/\***

**\* 说明：**

**\* 本来的想法是在原文件对象中覆盖内容，但是发现覆盖后文本为空， 无法解决**

**\* 但重新创建文件对象，则可以完成操作**

**\*/**

**File file1=new File("Message.txt");**

**if(!file1.exists())**

**file1.createNewFile();**

**FileOutputStream out=new FileOutputStream(file1,false);**

**out.write(sb1.toString().getBytes("gb2312"));**

**out.close();**

**return moneystring;**

**}**

**}**