高级语言程序设计I实验报告

**实验七 事件处理**

班 级： 22软件工程

姓 名： 王玉升

学 号： 22206091012

成 绩：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验名称 | | **实验七 事件处理** | | | |
| 实验日期 | | | 2023 | 实验课时 | 2 |
| 实  验  目  的  及  要  求 | （给出本次实验内容及要求的具体描述）   1. 实验目的：   （1）了解java.awt包和javax.swing包  （2）了解常用的容器、常用的布局、常用的组件  （3）理解事件处理模型  （4）掌握图形用户界面、事件驱动的应用程序设计，掌握窗口的创建、组件的添加、监视器的注册和事件的处理。  （5）掌握MVC结构应用程序的设计  （6）了解对话框的使用   1. 实验要求： 2. 安装**eclipse**开发环境，并使用eclipse环境进行java语言程序的编辑、编译和运行。 3. 在D盘或E盘以自己的学号末尾四位数字为名创建文件夹，并在其下面为每个实验项目创建子文件夹，程序相关文件存储在相应的文件夹中（如学号末尾是”1023”，实验六则应创建目录结构“E:\1023\sy06”，实验六程序文件放在“…\sy06”）。 4. 实验报告中附上程序清单和程序运行结果截图（每个窗口标题的后面显示自己的真实姓名）。  1. 每题程序代码中至少有两处用自己的真实姓名进行注释。   5．实验结束后进行总结。 | | | | |
| 实  验  环  境 | （列出本次实验所使用的平台和相关软件）  地点：南6207  开发环境：JDK1.8、eclipse | | | | |
| 实  验  过  程  及  实  验  结  果 | 1. **程序运行如下图所示：提供加运算练习，单击“获取题目”按钮，随机产生两个100以内的运算数，输入答案并按回车键或单击“确认答案”按钮，输出答案对错的信息。将【代码X】替换为Java程序代码（注释不能删除）：**   **/\*本题重点是掌握窗口的定义、随机数的生成、ActionEvent事件的处理、窗口作为监视器\*/**   * 1. 程序清单：   public class MainClass {  public static void main(String args[]) {  ComputerFrame frame;  frame=new ComputerFrame(); //创建窗口  frame.setTitle("加法运算练习－XXX"); //设置窗口标题, XXX为自己姓名  frame.setBounds(100,100,650,180); //设置窗口位置和大小 //王玉升  } }  import java.awt.\*; import java.awt.event.\*; import javax.swing.\*; import java.util.\*; public class ComputerFrame extends JFrame implements ActionListener{  JTextField textOne,textTwo,textResult;// 分别用于输出两个运算数和输入答案  JButton getProblem,giveAnwser;//两个按钮，分别用于出题和确定答案 王玉升  JLabel operatorLabel,message; //两个标签，分别用于显示运算符和结果是否正确  Random random; //用于产生两个随机运算数  int numberOne,numberTwo;//存放两两个随机产生的运算数  ComputerFrame() { //构造方法  setLayout(new FlowLayout()); //设置窗口布局方式为流式布局  random = new Random();  textOne = new JTextField(5);//【代码1】创建显示第一个运算数的文本框textOne,其可见字符长是5  textTwo=new JTextField(5); //创建显示第二个运算数的文本框textTwo  textResult=new JTextField(5); //创建输入练习者给出的答案的文本框textResult  operatorLabel=new JLabel("+"); //创建显示运算符“+”的标签operatorLabel  operatorLabel.setFont(new Font("Arial",Font.*BOLD*,20)); //设置运算符“+”的字体、字形与字号  message=new JLabel("你还没有回答呢"); //创建显示提示信息的标签message  getProblem=new JButton("获取题目"); //创建出题的命令按钮getProblem  giveAnwser=new JButton("确认答案"); //创建判断给出的答案对错的按钮giveAnwser  add(getProblem); //以下add()方法是将创建的各组件添加到窗口中  add(textOne);  add(operatorLabel);  add(textTwo);  add(operatorLabel);//【代码2】添加显示“＝”的标签对象  add(textResult);  add(giveAnwser); //王玉升  add(message);  textOne.setEditable(false);//第一文本框textOne不可操作  textTwo.setEditable(false);//【代码3】第二文本框textTwo不可操作  getProblem.setActionCommand("getProblem");//将getProblem命令按钮的"命令"字符串设为" getProblem "  textResult.setActionCommand("answer");//将textResult文本框的"命令"字符串设为"answer"  giveAnwser.setActionCommand("answer"); //将giveAnwser命令按钮的"命令"字符串设为"answer"  getProblem.addActionListener(this);//【代码4】为getProblem的ActionEvent事件监视器 ，窗口即为监视器  giveAnwser.addActionListener(this);//【代码5】为giveAnwser的ActionEvent事件监视器  textResult.addActionListener(this);//【代码6】为textResult的ActionEvent事件监视器  setVisible(true); //设置窗口显示  validate();  setDefaultCloseOperation(*DISPOSE\_ON\_CLOSE*); //设置窗口关闭方式  }  public void actionPerformed(ActionEvent e) {//重写actionPerformed方法  String str = e.getActionCommand(); //获取事件源的“命令”字符串  if(str.equals("getProblem")) {//如果事件源是"获取题目"按钮，也可通过e.getSource()来判断事件源是谁  numberOne = random.nextInt(100); //产生0至99之间的随机数作为第一运算数;  numberTwo = random.nextInt(100);//【代码7】产生0至99之间的随机数作为第二运算数;  textOne.setText(""+numberOne);//在textOne文本框显示第一个运算数  textTwo.setText("" + numberTwo);//【代码8】在textTwo文本框显示第二个运算数  message.setText("请回答"); //显示"请回答"的提示信息  textResult.setText(null); //输入答案文本框清空  textResult.requestFocus(); //让输出结果的文本框获得焦点  }  else if(str.equals("answer")) { ////如果事件源是输入答案文本框textResult或“确认答案”按钮giveAnwser  String answer = textResult.getText();//【代码9】获取textResult文本框文本（输入的答案）  try{ int result=Integer.*parseInt*(answer); //将输入的答案由文本型转换为整型  if(result==numberOne+numberTwo) //将判断输入的答案的对错  message.setText("你回答正确"); //显示答案正确与否的提示信息  else  message.setText("你回答错误"); //王玉升  }  catch(NumberFormatException ex) {  message.setText("请输入数字字符"); //如果输入的答案中含有非数字字符，给出提示信息  }  }  } }   * 1. 运行程序和运行结果（截图）：        1. **设计一程序，在下拉列表中选择学生，窗口中显示相应学生的具体信息，程序运行效果如图：**   **/\*本题重点是掌握下拉列表框的使用、ItemEvent事件处理、监视器类的定义\*/**  **要求：**   * **定义学生类Student:有表示学号、姓名和年龄的三个私有成员变量，一个有三个参数的构造方法，三个分别返回三个成员变量的方法。** * **定义窗口类WindowShow：一个Student类型数组表示创建一组学生对象（第一个人的姓名为自己的姓名），下拉列表中选项为各学生的姓名，创建监视器，为下拉列表框注册监视器，窗口中创建和添加组件，窗口可采用流式布局。** * **定义监视器ChoiceListener：有四个成员变量表示引用窗口中定义的三个文本框和存储学生信息的数组，四个方法分别设置这四个成员变量，重写接口方法itemStateChanged，实现将选择的学生信息显示在窗口文本框中** * **如果因JComboBox类的使用产生的警告提示，可忽略，原因参看第15章**  1. 程序清单：   import java.awt.\*; import java.awt.event.\*; import javax.swing.\*; import java.util.\*; import javax.swing.JComboBox; import javax.swing.JFrame; import javax.swing.JLabel; import javax.swing.JPanel; public class WindowShow extends JFrame {   JTextField IDOutput, NameOutput, AgeOutput;  JLabel ID, Name, Age;   JLabel giveName;  JComboBox ChooseStu;      public WindowShow(Student[] s) {  setLayout(new FlowLayout());  setVisible(true); //设置窗口显示  setDefaultCloseOperation(*DISPOSE\_ON\_CLOSE*); //设置窗口关闭方式      giveName = new JLabel("选择学生");  ChooseStu = new JComboBox<>();   for(int i = 0; i < 4; i++){  ChooseStu.addItem(s[i].getName());  }    ChooseStu.addItemListener(new ChoiceListener() {  @Override  public void itemStateChanged(ItemEvent e) { // if(e.getSource() == )     }  });      ID = new JLabel("学号");  Name = new JLabel("姓名");  Age = new JLabel("年龄");   IDOutput = new JTextField(5);  NameOutput = new JTextField(5);  AgeOutput = new JTextField(5);   add(giveName);  add(ChooseStu);   add(ID);  add(IDOutput);  add(Name);  add(NameOutput);  add(Age);  add(AgeOutput);    }     public static void main(String[] args) {   Student Infor[] = new Student[4]; //定义学生数组  Infor[0] = new Student(1012, 20, "王玉升");  Infor[1] = new Student(458, 20, "陈丽娜");  Infor[2] = new Student(732, 20, "张伟杰");  Infor[3] = new Student(891, 20, "李婷婷");   WindowShow s = new WindowShow(Infor);  s.setBounds(100,100,800,410);       }    }  import javax.swing.\*; import java.awt.event.ActionEvent; import java.awt.event.ActionListener; import java.awt.TextField; import java.awt.event.\*; import java.awt.\*;  public abstract class ChoiceListener implements ItemListener {   private JTextField NameTextField;  private JTextField IDTextField;  private JTextField AgeTextField;  private Choice StudentChoice;  private String[][] studentData;   public ChoiceListener(JTextField NameTextField, JTextField IDTextField, JTextField AgeTextField, String[][] studentData) {  this.NameTextField = NameTextField;  this.IDTextField = IDTextField;  this.AgeTextField = AgeTextField;  this.studentData = studentData;  }   public ChoiceListener() {   }    public void setNameTextField(JTextField nameTextField) {  this.NameTextField = nameTextField;  }   // 方法2：设置IDTextField  public void setIdTextField(JTextField idTextField) {  this.IDTextField = idTextField;  }   // 方法3：设置AgeTextField  public void setGradeTextField(JTextField gradeTextField) {  this.AgeTextField = gradeTextField;  } }  （2） 运行程序和运行结果（截图）：     1. **编程：用匿名类对象作监视器实现第2题（选做）**   **/\*本题重点是掌握匿名类对象作监视器\*/**   * 1. 程序清单：   2. 运行程序和运行结果（截图）：  1. **设计一用于登录的程序，当输入用户名为user01密码为123时，在消息窗口显示”登录成功，否则显示“用户名或密码错误!”,程序运行如下图所示：**     **/\*本题重点是掌握ActionEvent事件处理、对话的使用\*/**   * 1. 程序清单：   **Sy6\_4.java：**  //主类  public class Sy6\_4 {  public static void main(String args[]) {  WindowLogin win=new WindowLogin();  win.setTitle(" 登录窗口－XXX");  win.setBounds(200,150,250,200);  }  } **WindowLogin.java：**  //定义窗口类，且窗口作监视器  //定义窗口类，且窗口作监视器 import java.awt.\*; import java.awt.event.\*; import javax.swing.\*; import javax.swing.border.\*; public class WindowLogin extends JFrame implements ActionListener {  JTextField textName;// 用于输入用户名  JPasswordField textPassword;// 用于输入密码  JButton button1,button2;//两个按钮，分别用于确定和取消输入  WindowLogin() {  setLayout(new FlowLayout()); //【代码1】设置窗口布局方式为流式布局  textName = new JTextField(10); //创建文本框textName,其可见字符长是10  textPassword = new JPasswordField(10);//【代码2】 创建密码框textPassword,其可见字符长是10  button1 = new JButton("确定");//【代码3】 创建“确定”按钮button1  button2=new JButton("取消");  add(new JLabel(" 用户名："));  add(textName);//【代码4】 窗口添加textName文本框  add(new JLabel(" 密 码："));  add(textPassword);  add(button1);  add(button2);  textName.requestFocus();  button1.addActionListener(this);//【代码5】为button1注册ActionEvent事件监视器  button2.addActionListener(this);//【代码6】为button2注册ActionEvent事件监视器  setVisible(true);//【代码7】窗口可见  validate();  setDefaultCloseOperation(*DISPOSE\_ON\_CLOSE*);//【代码8】窗口关闭方式为DISPOSE\_ON\_CLOSE  }  public void actionPerformed(ActionEvent e) {//重写actionPerformed方法  if(e.getSource()==button1) {  String name = new String(textName.getName());//【代码9】获取用户名  String password = new String(textPassword.getPassword());//获取密码  if(name.equals("user01") && password.equals("123")){  JOptionPane.*showMessageDialog*(this,"登录成功！","消息对话框", JOptionPane.*WARNING\_MESSAGE*); //在消息窗口显示"登录成功！"  }  else{  JOptionPane.*showMessageDialog*(this, "用户名或密码错误!");//【代码10】在消息窗口显示"用户名或密码错误！"  }  }  else if(e.getSource()==button2) {  textName.setText(null); //用户名框清空  textPassword.setText(null);//【代码11】将密码框清空  }  } }   * 1. }运行程序和运行结果（截图）：  1. **完善第5题的登录验证程序，添加验证码功能，验证码由随机生成的四位数字组成，当输入的用户名、密码或验证码有错误时，在消息窗口显示相应的错误信息，否则在消息窗口显示“登录成功”，程序运行如下图所示（正确的用户名为user01，密码为123，选做）：**       **/\*本题重点是掌握ActionEvent事件处理、随机数、对话框的使用\*/**   * 1. 程序清单：   2. 运行程序和运行结果（截图）：  1. **编写一个体现MVC结构的GUI程序，首先编写一个封装梯形类，然后再编写一个窗口，要求窗口使用三个文本框和一个文本区为梯形中的数据提供视图，其中三个文本框用来显示梯形的上底、下底和高，文本区用来显示梯形的面积，窗口中有一按钮，单击按钮后用文本框中的数据作为梯形的上底、下底和高，计算梯形面积，面积显示在文本区。运行结果如图所示：**     **/\*本题重点是掌握MVC结构的GUI程序的设计\*/**   * 1. 程序清单：   2. 运行程序和运行结果（截图）：  1. **编写一个以树形式显示各省及城市的程序，运行结果如图所示（选做）：**     **程序要求：**  **（1）选择树中某节点，右边文本区显示“选择的是XXX”的信息。**  **（2）在文本框输入新节点标签后，单击“添加”按钮，将为选择的节点添加一子节点，子节点的标签为文本框输入的内容，并在右边文本区显示“XXX节点已插入！”的信息**  **（3）单击“删除”按钮，将为选择的节点删除，并在右边文本区显示“XXX节点已删除！”的信息**   * 1. 程序清单：   2. 运行程序和运行结果（截图）：  1. **编写一个工资单编辑程序，工资单基本数据如下（选做）：**  |  |  |  |  |  | | --- | --- | --- | --- | --- | | **工号** | **姓名** | **性别** | **年龄** | **工资** | | 1001 | 曹平威 | 男 | 30 | 5500 | | 1002 | 陈华娟 | 女 | 28 | 4300 | | 1003 | 胡晓晓 | 女 | 32 | 4800 | | 1004 | 胡先强 | 男 | 26 | 4250 | | 1005 | 程刚鹏 | 男 | 45 | 6300 | | 1006 | 戴林林 | 男 | 36 | 5000 |   **运行结果如图所示：**    **程序要求：**  **（1）表格中数据可以插入、删除和修改。**  **（2）表格只能单行选择，选择的行背景色为RGB(100,180,150),前景色为红色。**  **（3）单击“添加”按钮，在表格末尾添加新的空白行，可录入数据。**  **（4）单击“删除”按钮，删除表格中选择的行**   * 1. 程序清单：   2. 运行程序和运行结果（截图）： | | | | |
| 总  结 | 实验心得体会： | | | | |