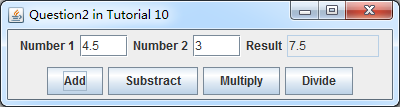
**《Java程序设计》上机作业7**

|  |  |
| --- | --- |
| **学号：** | **202103151422** |
| **姓名：** | **温家伟** |
| **班级：** | **大数据分析2101** |
| **日期：** | **2023-12-23** |

1. (*Creating a simple calculator*) 编写一个能进行简单四则运算的计算器程序，用户输入两个运算数之后，点击add, subtract, multiply, 和 divide 按钮可以分别执行加减乘除运算，并在Result区域显示运算结果。如下图所示：

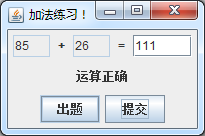


|  |
| --- |
| * **问题分析与解答**   要编写一个简单的四则运算计算器程序，需要实现以下功能：  用户输入两个运算数。  用户点击加、减、乘、除按钮执行相应的运算。  将运算结果显示在结果区域。 |
| * **代码实现**   package test10;  import java.awt.event.\*;  import java.awt.\*;  import javax.swing.\*;  public class cal {  public static void main(String args[]) {  JFrame frm = new JFrame("Question2 in Tutorial 10");  frm.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  frm.setSize(400, 120);  Container c = frm.getContentPane();  c.setLayout(new FlowLayout());  JTextField[] t = {new JTextField("", 5),  new JTextField("", 5),  new JTextField("", 5)};  c.add(new JLabel("Number 1"));  c.add(t[0]);  c.add(new JLabel("Number 2"));  c.add(t[1]);  c.add(new JLabel("Result"));  c.add(t[2]);  t[2].setEditable(false); // 结果设置为只读  JButton b1 = new JButton("Add");  JButton b2 = new JButton("Substract");  JButton b3 = new JButton("Multiply");  JButton b4 = new JButton("Divide");  b1.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  try{  double num1 = Double.parseDouble(t[0].getText());  double num2 = Double.parseDouble(t[1].getText());  double result = num1 + num2;  t[2].setText(Double.toString(result));  }catch(NumberFormatException ex){  JOptionPane.showMessageDialog(frm,"输入格式错误！","警告",JOptionPane.WARNING\_MESSAGE);  }  }  });  b2.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  try{  double num1 = Double.parseDouble(t[0].getText());  double num2 = Double.parseDouble(t[1].getText());  double result = num1 - num2;  t[2].setText(Double.toString(result));  }catch(NumberFormatException ex){  JOptionPane.showMessageDialog(frm,"输入格式错误！","警告",JOptionPane.WARNING\_MESSAGE);  }  }  });  b3.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  try{  double num1 = Double.parseDouble(t[0].getText());  double num2 = Double.parseDouble(t[1].getText());  double result = num1 \* num2;  t[2].setText(Double.toString(result));  }catch(NumberFormatException ex){  JOptionPane.showMessageDialog(frm,"输入格式错误！","警告",JOptionPane.WARNING\_MESSAGE);  }  }  });  b4.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  try{  double num1 = Double.parseDouble(t[0].getText());  double num2 = Double.parseDouble(t[1].getText());  if(num2 == 0){  JOptionPane.showMessageDialog(frm,"除数不能为0！","警告",JOptionPane.WARNING\_MESSAGE);  }else{  double result = num1 / num2;  String resultStr = String.format("%.2f", result);  t[2].setText(resultStr);  }  }catch(NumberFormatException ex){  JOptionPane.showMessageDialog(frm,"输入格式错误！","警告",JOptionPane.WARNING\_MESSAGE);  }  }  });  c.add(b1);  c.add(b2);  c.add(b3);  c.add(b4);  frm.setVisible(true);  }  } |
| * **运行结果**     69d46fa3db050519964d143d00f4936  e889ed8a2bb93e0a116d08787c8b57a  300b8e35d826a3acdf1faf1f145a0e4 |
| * **思考及总结**   我们可以开发一个简单的四则运算计算器程序，能够满足用户输入运算数并执行加、减、乘、除运算的需求。但需要注意处理用户可能输入的非法字符和除数为0的情况，以及将运算结果显示在界面上。 |

1. 编写一个图形用户界面程序。界面包括两个按钮（JButton对象）、三个输入框（JTextField对象）和三个标签（JLabel对象）。其中：两个输入框不可编辑，三个标签分别是“+”、“=”和位于中间区域的提示信息。

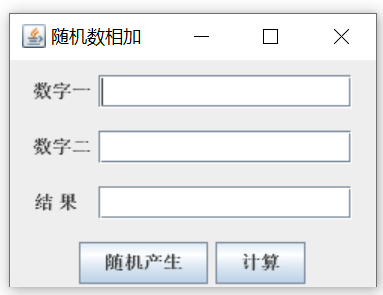
- 用户点击“出题”按钮，程序随时生成两个[0, 100]范围内的整数显示在两个不可编辑输入框中；

- 用户点击“提交”按钮，程序比对第三个输入框内的数值与随机生成的两个整数的和，如果相等，则提示信息为“运算正确”，如不相等，则提示“运算错误”，如未输入任何信息则提示“输入有误”，如未曾出题，则提示“请先出题”。程序界面如图所示：



|  |
| --- |
| * **问题分析与解答**   这是一个简单的界面交互程序，需要实现以下功能：  生成两个随机整数  在界面上显示这两个整数的加法运算  用户输入计算结果  判断用户回答是否正确  在界面上给出相应的提示  可以将这个程序分解成以下四个部分来实现：  界面设计：使用 Swing 库创建窗口和控件  随机数生成：使用 Random 类生成两个随机整数  加法运算：将两个随机整数相加得到正确答案  用户交互：处理用户输入和判断用户回答是否正确 |
| * **代码实现**   package test10;  import javax.swing.\*;  import java.awt.event.ActionEvent;  import java.awt.event.ActionListener;  import java.util.Random;  public class exerciseAdd extends JFrame {  private JTextField num1Field, num2Field, resultField, infoLabel;  public exerciseAdd() {  setTitle("加法练习！");  setSize(300, 150);  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  JButton generateBtn = new JButton("出题");  JButton submitBtn = new JButton("提交");  JLabel plusLabel = new JLabel("+");  JLabel equalsLabel = new JLabel("=");  num1Field = new JTextField(5);  num1Field.setEditable(false);  num2Field = new JTextField(5);  num2Field.setEditable(false);  resultField = new JTextField(5);  infoLabel = new JTextField(25);  infoLabel.setEditable(false);  infoLabel.setBorder(null);  infoLabel.setHorizontalAlignment(JTextField.CENTER);  JPanel panel = new JPanel();  panel.add(num1Field);  panel.add(plusLabel);  panel.add(num2Field);  panel.add(equalsLabel);  panel.add(resultField);  panel.add(infoLabel);  panel.add(generateBtn);  panel.add(submitBtn);  add(panel);  generateBtn.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  Random rand = new Random();  int num1 = rand.nextInt(101);  int num2 = rand.nextInt(101);  num1Field.setText(String.valueOf(num1));  num2Field.setText(String.valueOf(num2));  }  });  submitBtn.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  if (num1Field.getText().isEmpty() || num2Field.getText().isEmpty()) {  infoLabel.setText("请先出题");  } else if (resultField.getText().isEmpty()) {  infoLabel.setText("输入有误");  } else {  int num1 = Integer.parseInt(num1Field.getText());  int num2 = Integer.parseInt(num2Field.getText());  int result = Integer.parseInt(resultField.getText());  if (result == num1 + num2) {  infoLabel.setText("运算正确");  } else {  infoLabel.setText("运算错误");  }  }  }  });  }  public static void main(String[] args) {  exerciseAdd gui = new exerciseAdd();  gui.setVisible(true);  }  } |
| * **运行结果** |
| * **思考及总结**   当用户点击提交按钮时，程序需要从输入框中读取用户输入的计算结果，并与正确答案进行比较。如果用户回答正确，则在界面上显示“运算正确”，否则显示“运算错误”。 |

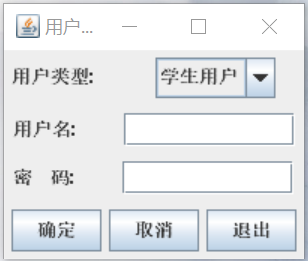
1. 设计如下图所示的图形用户界面，随机生成两个100以内的整数，并对这两个数进行求和。按“随机产生”按钮，产生两个随机数并分别显示在“数字一”和“数字二”对应的文本框内，按“计算”按钮，计算产生的这两个随机数之和，并显示在“结果”对应的文本框内。



|  |
| --- |
| * **问题分析**   界面设计：需要使用 Swing 库创建窗口和控件来实现图形用户界面。界面上应该包含以下组件：  两个文本框用于显示随机生成的整数。  一个按钮用于生成随机数。  一个按钮用于计算两个随机数的和。  一个文本框用于显示计算结果。  随机数生成：需要使用 Random 类生成两个随机整数，并将其显示在对应的文本框中。生成的随机整数应该在 0 到 100 之间。  加法运算：需要获取两个随机整数的值，并将它们相加得到正确答案。正确答案应该是一个整数。  用户交互：需要处理用户点击按钮的事件，并做出相应的操作。当用户点击"生成随机数"按钮时，程序应生成两个随机整数并将其显示在文本框中；当用户点击"计算"按钮时，程序应获取两个随机整数的值，并计算它们的和，并将结果显示在文本框中。 |
| * **代码实现**   **package test10;**  **import javax.swing.\*;**  **import java.awt.\*;**  **import java.awt.event.ActionEvent;**  **import java.awt.event.ActionListener;**  **import java.util.Random;**  **public class MathQuizGUI extends JFrame {**  **private JTextField num1Field, num2Field, resultField;**  **public MathQuizGUI() {**  **setTitle("随机数相加");**  **setSize(400, 160);**  **setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);**  **JButton generateBtn = new JButton("随机产生");**  **JButton calculateBtn = new JButton("计算");**  **JLabel num1Label = new JLabel("数字一:");**  **JLabel num2Label = new JLabel("数字二:");**  **JLabel resultLabel = new JLabel("结 果: ");**  **num1Field = new JTextField(28);**  **num2Field = new JTextField(28);**  **resultField = new JTextField(28);**  **JPanel panel = new JPanel();**  **panel.add(num1Label);**  **panel.add(num1Field);**  **panel.add(num2Label);**  **panel.add(num2Field);**  **panel.add(resultLabel);**  **panel.add(resultField);**  **panel.add(generateBtn);**  **panel.add(calculateBtn);**  **add(panel);**  **generateBtn.addActionListener(new ActionListener() {**  **@Override**  **public void actionPerformed(ActionEvent e) {**  **Random rand = new Random();**  **int num1 = rand.nextInt(101);**  **int num2 = rand.nextInt(101);**  **num1Field.setText(String.valueOf(num1));**  **num2Field.setText(String.valueOf(num2));**  **}**  **});**  **calculateBtn.addActionListener(new ActionListener() {**  **@Override**  **public void actionPerformed(ActionEvent e) {**  **int num1 = Integer.parseInt(num1Field.getText());**  **int num2 = Integer.parseInt(num2Field.getText());**  **int result = num1 + num2;**  **resultField.setText(String.valueOf(result));**  **}**  **});**  **}**  **public static void main(String[] args) {**  **MathQuizGUI gui = new MathQuizGUI();**  **gui.setVisible(true);**  **}**  **}** |
| * **运行结果** |
| * **思考及总结**   这段代码使用了 Swing 库创建了一个窗口，并添加了标签、文本框和按钮。点击"随机产生"按钮会生成两个随机整数，并分别显示在"数字一"和"数字二"对应的文本框内。点击"计算"按钮会将这两个随机数相加，并将结果显示在"结果"对应的文本框内。每个按钮都有对应的事件处理程序，用于处理按钮点击事件。 |

1. 编写一个用户登陆程序UserLoginApp.java，具体要求如下：

① 用户界面大小为200\*160像素，如下图所示



用户登录程序

② 用户类型包括学生用户和教师用户，缺省为学生用户。

③ “确定”按钮功能：如果用户名为空，则打印"用户名不可为空!"；如果密码为空，则打印"密码不可为空!"；如果是学生用户，用户名和密码都是stu，登陆成功则打印"学生用户登陆成功"；如果是教师用户，用户名和密码都是tea；登陆成功则打印"教师用户登陆成功"；如果登陆不成功，则打印"用户名不存在或者密码不正确!"。

④ “取消”按钮功能：将用户名和密码的输入框清空。

⑤ “退出”按钮功能：退出程序

提高要求：教师和学生的用户名、密码、用户类型等都保存在user.txt文件中，登录时从文件中读取用户名密码并进行判断。文档内容具体格式如下：

张同学 123456 student

李同学 123123 student

王老师 456456 teacher

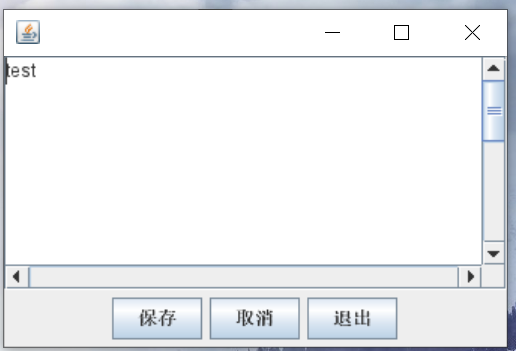
|  |
| --- |
| * **问题分析与解答**   用户希望实现一个用户登录程序，要求用户输入用户名、密码和用户类型，并根据输入的信息判断登录是否成功。 |
| * **代码实现**   package test10;  import javax.swing.\*;  import java.awt.\*;  import java.awt.event.ActionEvent;  import java.awt.event.ActionListener;  import java.io.BufferedReader;  import java.io.FileReader;  import java.io.IOException;  public class UserLoginApp extends JFrame {  private JTextField usernameField, passwordField;  private JComboBox<String> userTypeComboBox;  private JButton loginBtn, cancelBtn, exitBtn;  public UserLoginApp() {  setTitle("用户登录程序");  setSize(280, 160);  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  JLabel usernameLabel = new JLabel("用户名:");  JLabel passwordLabel = new JLabel("密 码: ");  JLabel space = new JLabel(" ");  usernameField = new JTextField(15);  passwordField = new JTextField(15);  userTypeComboBox = new JComboBox<>(new String[]{"学生用户", "教师用户"});  loginBtn = new JButton("确定");  cancelBtn = new JButton("取消");  exitBtn = new JButton("退出");  JPanel panel = new JPanel();  panel.add(new JLabel("用户类型: "));  panel.add(userTypeComboBox);  panel.add(space);  panel.add(usernameLabel);  panel.add(usernameField);  panel.add(passwordLabel);  panel.add(passwordField);  panel.add(loginBtn);  panel.add(cancelBtn);  panel.add(exitBtn);  add(panel);  // 确定按钮事件处理  loginBtn.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  String username = usernameField.getText();  String password = passwordField.getText();  String userType = (String) userTypeComboBox.getSelectedItem();  if (username.isEmpty()) {  JOptionPane.showMessageDialog(null, "用户名不可为空!");  } else if (password.isEmpty()) {  JOptionPane.showMessageDialog(null, "密码不可为空!");  } else {  boolean loginSuccess = false;  try (BufferedReader reader = new BufferedReader(new FileReader("D:\\编程\\阿伟的笔记本\\大三上\\java\\作业\\作业7\\user.txt"))) {  String line;  while ((line = reader.readLine()) != null) {  String[] user = line.split(" ");  String storedUsername = user[0];  String storedPassword = user[1];  String storedUserType = user[2];  if (username.equals(storedUsername) && password.equals(storedPassword)) {  userType = storedUserType;  loginSuccess = true;  break;  }  }  } catch (IOException ex) {  ex.printStackTrace();  }  if (loginSuccess) {  if (userType.equals("student")) {  JOptionPane.showMessageDialog(null, "学生用户登录成功");  } else if (userType.equals("teacher")) {  JOptionPane.showMessageDialog(null, "教师用户登录成功");  }  } else {  JOptionPane.showMessageDialog(null, "用户名不存在或密码不正确!");  }  }  }  });  cancelBtn.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  usernameField.setText("");  passwordField.setText("");  }  });  exitBtn.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  System.exit(0);  }  });  }  public static void main(String[] args) {  UserLoginApp app = new UserLoginApp();  app.setVisible(true);  }  } |
| * **运行结果** |
| * **思考及总结**   我们可以理解用户登录程序的基本逻辑，并对界面设计、用户输入验证、文件操作和布局管理器等知识点有了更深入的了解。同时，在实现具体功能时，需要仔细处理各种异常情况，以提高程序的稳定性和用户体验。 |

1. 使用Swing组件编写一个支持中文文本编辑程序ChineseTextEdit.java，要求如下：
2. 用户界面大小为400×200像素，如下图所示：
3. 程序启动后，多行文本输入框JTextArea中显示当前目录下myText.txt文件中原有的内容，如果该文件不存在，则新建该文件。
4. “保存”按钮功能：将多行文本输入框JTextArea中的内容写入myText.txt文件中保存。
5. “取消”按钮功能：将多行文本输入框TextArea中的内容清空。
6. “退出”按钮功能：退出程序。

程序启动后，多行文本输入框JTextArea中显示当前目录下myText.txt文件内容。

*提高要求（选做）：*

1. 保存按钮点击后，弹出保存文件对话框（FileDialog/JFileChooser），可以编辑文件名保存到指定目录：
2. 退出按钮点击后，如果文档内容已保存则直接退出，如果未保存，则弹出对话框进行二次确认（确定/取消），以确认退出程序或者取消操作。



|  |
| --- |
| * **问题分析与解答**   这是一道要求使用Swing组件编写一个支持中文文本编辑程序的题目，主要涉及到如何使用Swing组件来实现用户界面的设计和文件读写操作。具体要求如下：用户界面大小为400×200像素，程序启动后，多行文本输入框JTextArea中显示当前目录下myText.txt文件中原有的内容，如果该文件不存在，则新建该文件。“保存”按钮功能：将多行文本输入框JTextArea中的内容写入myText.txt文件中保存。“取消”按钮功能：将多行文本输入框TextArea中的内容清空。“退出”按钮功能：退出程序。 |
| * **代码实现**   package test10;  import javax.swing.\*;  import java.awt.\*;  import java.awt.event.ActionEvent;  import java.awt.event.ActionListener;  import java.io.\*;  public class ChineseTextEdit extends JFrame {  private JTextArea textArea;  public ChineseTextEdit() {  setTitle("中文文本编辑程序");  setSize(400, 200);  setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  textArea = new JTextArea();  loadFileContent();  JScrollPane scrollPane = new JScrollPane(textArea);  JButton saveButton = new JButton("保存");  JButton clearButton = new JButton("取消");  JButton exitButton = new JButton("退出");  saveButton.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  saveFileContent();  }  });  clearButton.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  textArea.setText("");  }  });  exitButton.addActionListener(new ActionListener() {  @Override  public void actionPerformed(ActionEvent e) {  confirmExit();  }  });  getContentPane().setLayout(new BorderLayout());  getContentPane().add(scrollPane, BorderLayout.CENTER);  JPanel buttonPanel = new JPanel();  buttonPanel.add(saveButton);  buttonPanel.add(clearButton);  buttonPanel.add(exitButton);  getContentPane().add(buttonPanel, BorderLayout.SOUTH);  }  private void loadFileContent() {  try (BufferedReader reader = new BufferedReader(new FileReader("myText.txt"))) {  String line;  StringBuilder content = new StringBuilder();  while ((line = reader.readLine()) != null) {  content.append(line).append("\n");  }  textArea.setText(content.toString());  } catch (IOException e) {  // 文件不存在则创建新文件  File file = new File("myText.txt");  try {  file.createNewFile();  } catch (IOException ex) {  ex.printStackTrace();  }  }  }  private void saveFileContent() {  JFileChooser fileChooser = new JFileChooser();  int result = fileChooser.showSaveDialog(this);  if (result == JFileChooser.APPROVE\_OPTION) {  File selectedFile = fileChooser.getSelectedFile();  try (BufferedWriter writer = new BufferedWriter(new FileWriter(selectedFile))) {  writer.write(textArea.getText());  } catch (IOException e) {  e.printStackTrace();  }  }  }  private void confirmExit() {  int result = JOptionPane.showConfirmDialog(this, "确定退出程序吗？", "确认退出", JOptionPane.YES\_NO\_OPTION);  if (result == JOptionPane.YES\_OPTION) {  System.exit(0);  }  }  public static void main(String[] args) {  ChineseTextEdit app = new ChineseTextEdit();  app.setVisible(true);  }  } |
| * **运行结果** |
| * **思考及总结**   本题要求使用Swing组件实现文本编辑器的功能，需要掌握Swing组件的常用方法和事件处理机制。在代码实现过程中，需要注意以下几点：  创建并设置JFrame窗口的属性，包括大小、位置、关闭操作等。  创建JTextArea和JScrollPane组件，并将JTextArea添加到JScrollPane组件中。  判断当前目录下是否存在myText.txt文件，如果不存在则创建该文件。  使用FileReader和BufferedReader读取指定文件的内容，并将内容显示到JTextArea中。  创建JButton组件并添加事件监听器，在事件处理方法中实现文件保存、清空、退出等功能。  保存文件时可以使用FileDialog或JFileChooser对话框来选择保存路径和文件名。  退出程序时可以根据文件是否保存弹出确认对话框。  总之，要实现一个完整的文本编辑器，需要综合运用Swing组件、文件读写、事件处理等知识。 |