**重庆交通大学信息科学与工程学院**

**综合性设计性实验报告**

**班 级： 计科1703**

**姓名 学号： 黄富灵 631707060308**

**实验项目名称： 实验七、GUI编程**

**实验项目性质： 综 合 性**

**实验所属课程： 《Java程序设计》**

**实验室(中心)：**

**指 导 教 师 ： 王桂平**

**实验完成时间： 2019年 3月 31 日**

教师评阅意见：

签名： 年 月 日

实验成绩：

**一、实验题目**

采用Java语言和相关技术编写一个GUI程序，要求尽可能使用Java GUI中的容器、布局管理器、各种组件、事件处理机制、菜单、绘图等技术。

**二、实验要求**

1) 分组要求：分组完成，自由组队，2人一组。

2) 程序要求：不少于150行代码，且具有较好的可读性（注释、正确缩进等）。

3) 文档要求：详细阐述该程序的设计思想、采用的Java技术或方法。

**三、设计思想、采用的Java技术或方法**

整体的页面布局可设为空布局，上下各一个面板，其中上面面板可放入文本输入域（jtf.setHorizontalAlignment (JTextField.RIGHT);）这句代码是将文本件右边往左边输出，下面则是放入 button，中间有个分隔线

功能主要是实现了基本的加减乘除，至于加减则是则是从文本输入域中获取要操作的数据，其中要注意一点，加减乘除是特殊字符（减法除外），要通过转译才能，例如加法（"\\+"），要使用这样的操作才能运行，否则会在控制台输出错误异常信息 java.util.regex.PatternSyntaxException

**四、主要程序代码**

*/\**

*\* @Coding: utf-8 \n*

*\* @Despcription: \n*

*\* @Author: Suwen SP \n*

*\* @LastEditors: Suwen SP\n*

*\* @Date: 2019-03-31 15:54:10*

*\* @LastEditTime: 2019-03-31 15:59:11*

*\*/*

import java.awt.BasicStroke;

import java.awt.Color;

import java.awt.Font;

import java.awt.Graphics;

import java.awt.Graphics2D;

import java.awt.GridLayout;

import javax.swing.BorderFactory;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JTextField;

import javax.swing.UIManager;

public class Calc extends JFrame {

private static final long serialVersionUID = 1L;

private void initFrame() {

try {

UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());

} catch (Exception *e*) {

e.printStackTrace();

}

this.setSize(300, 400);

this.setTitle("SIMPLE\_CALC");

this.setLocationRelativeTo(null);

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.setLayout(null);

*// JTextField调用方法时传递*

JTextField jtf = this.topSurface();

this.bottomSurface(jtf);

this.setVisible(true);

}

private JTextField topSurface() {

JPanel topPanel = new JPanel();

topPanel.setBounds(-5, 10, 300, 50);

JTextField jtf = new JTextField(16);

*// 设置文件右边往左边输出*

jtf.setHorizontalAlignment(JTextField.RIGHT);

jtf.setForeground(Color.red);

jtf.setFont(new Font("SansSerif", Font.PLAIN, 20));

jtf.setEditable(false);

jtf.setText("0");

topPanel.add(jtf);

this.add(topPanel);

return jtf;

}

private void bottomSurface(JTextField *jtf*) {

JPanel bottomPanel = new JPanel();

bottomPanel.setBounds(8, 80, 270, 250);

bottomPanel.setLayout(new GridLayout(5, 4, 1, 1));

String[] buttonValue = { "1", "2", "3", "C", "4", "5", "6", "CE", "7", "8", "9", "←", "+", "-", "0", ".", "\*",

"/", "%", "=" };

*// 监听器不能重复创建*

CalcListener cl = new CalcListener(jtf);

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

JButton button = new JButton(buttonValue[i]);

button.setBorder(BorderFactory.createRaisedBevelBorder());

button.setFont(new Font("宋体", 0, 25));

button.setForeground(Color.blue);

bottomPanel.add(button);

button.addActionListener(cl);

}

this.add(bottomPanel);

}

*/\*\**

*\* =============================计算器分割线========================================*

*\*\*/*

*// 定义自己的画笔，把paint方法的Graphics对象强转为自定义画笔*

private Graphics2D g;

@Override

public void paint(Graphics *g*) {

super.paint(g);

this.g = (Graphics2D) g;

this.g.setStroke(new BasicStroke(3));

this.g.setColor(new Color(18, 18, 9));

this.g.drawLine(0, 95, this.getWidth(), 95);

}

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

Calc calc = new Calc();

calc.initFrame();

}

}

*/\**

*\* @Coding: utf-8 \n*

*\* @Despcription: \n*

*\* @Author: Suwen SP \n*

*\* @LastEditors: Suwen SP\n*

*\* @Date: 2019-03-31 15:55:40*

*\* @LastEditTime: 2019-03-31 15:57:28*

*\*/*

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JTextField;

public class CalcListener implements ActionListener {

private JTextField jtf;

*// 创建字符缓冲区域*

String content = "";

private StringBuffer sb = new StringBuffer();

public CalcListener(JTextField *jtf*) {

this.jtf = jtf;

}

@Override

public void actionPerformed(ActionEvent *e*) {

String command = e.getActionCommand();

switch (command) {

*// case同时写多个条件，数字的基本处理*

case "1":

case "2":

case "3":

case "4":

case "5":

case "6":

case "7":

case "8":

case "9":

case "0":

case ".":

case "+":

case "-":

case "\*":

case "/":

case "%":

sb.append(command);

jtf.setText(sb.toString().trim());

break;

*// 清空输入框中内容*

case "C":

case "CE":

*// 清空StringBuffer*

sb.setLength(0);

jtf.setText("0");

break;

*// 退格*

case "←":

break;

*// 四则运算处理 处理两个运算符问题两个点*

case "=":

String content = jtf.getText();

if (content.contains("+")) {

String[] string = content.split("\\+");

String frist = string[0];

String next = string[1];

double result = Double.valueOf(frist) + Double.valueOf(next);

jtf.setText(String.valueOf(result));

} else if (content.contains("-")) {

String[] string = content.split("-");

String frist = string[0];

String next = string[1];

double result = Double.valueOf(frist) - Double.valueOf(next);

jtf.setText(String.valueOf(result));

} else if (content.contains("\*")) {

String[] string = content.split("\\\*");

String frist = string[0];

String next = string[1];

double result = Double.valueOf(frist) \* Double.valueOf(next);

jtf.setText(String.valueOf(result));

} else if (content.contains("/")) {

String[] string = content.split("\\/");

String frist = string[0];

String next = string[1];

double result = Double.valueOf(frist) / Double.valueOf(next);

jtf.setText(String.valueOf(result));

} else if (content.contains("%")) {

String[] string = content.split("\\%");

String frist = string[0];

String next = string[1];

double result = Double.valueOf(frist) % Double.valueOf(next);

jtf.setText(String.valueOf(result));

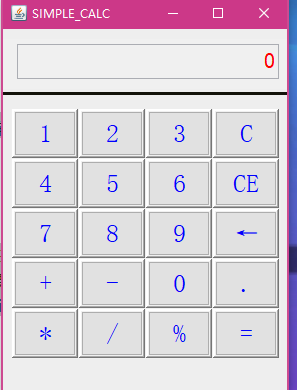
}

break;

}

}

}

**五、测试结果及说明**

**六、实验体会**

Swing 是一种 java 语言开发窗口程序的工具包，是以抽象的工具类 AWT (Abstract WindowsToolkit) 为基础发展而来，由于 AWT 的一些重要功能还有所欠缺，于是就出现了 Swing，但并不是抛弃了 AWT，一些重要的工具仍然保存，但是 Swing 在选择上还是成为了开发的首选。

图形用户界面是一组图形界面成分和界面元素的有机组合。这些成分和元素之间不仅外观上有包含、相邻、相交等物理关系，内在的也包含、调用等逻辑关系。他们相互作用，传递消息，共同组成一个能响应特定事件的、具有一定功能的图形界面系统。