**学习总结**

姓名 董金伟 专业 计算机科学与技术 班级 2班 日期 2022/4/8

1. 当天所学习主要内容（做一总结）

# 菜单

菜单可以直接放到窗体中

图形用户界面, 文本, 应用程序

描述已自动生成

菜单的组成：菜单条，菜单，菜单项

菜单添加到窗体中的步骤：

1. 把菜单项放到菜单中
2. 把菜单放到菜单条中
3. 把菜单条放到窗体中

public class MyFrame extends JFrame {

//菜单条

private JMenuBar jMenuBar;

//菜单

private JMenu jMenu1,jMenu2,jMenu3;

//菜单项

private JMenuItem jMenuItem11,jMenuItem12,jMenuItem13;

public MyFrame(){

//创建相关对象

jMenuBar = new JMenuBar();

jMenu1 = new JMenu("文件");

jMenu2 = new JMenu("编辑");

jMenu3 = new JMenu("格式");

jMenuItem11 = new JMenuItem("新建");

jMenuItem12 = new JMenuItem("新窗口");

jMenuItem13 = new JMenuItem("打开");

// 1）把菜单项放到菜单中

jMenu1.add(jMenuItem11);jMenu1.add(jMenuItem12);jMenu1.add(jMenuItem13);

// 2）把菜单放到菜单条中

jMenuBar.add(jMenu1);jMenuBar.add(jMenu2);jMenuBar.add(jMenu3);

// 3）把菜单条放到窗体中

this.add(jMenuBar, BorderLayout.NORTH);

this.setTitle("测试菜单");

this.setSize(300,500);

this.setLocationRelativeTo(null);

this.setVisible(true);

this.setDefaultCloseOperation(EXIT\_ON\_CLOSE);

}

}

# 布局（把组件放到指定的位置）

1. BorderLayout(边框布局)，可以把组件放到东(EAST)南(SOUTH)西(WEST)北(NORTH)中(CENTER)

JFrame(窗体)默认使用的是边框布局，如果不指定方向时，默认在中间

图形用户界面, 文本, 应用程序

描述已自动生成

1. 流式布局（FlowLayout）:按行进行摆放，如果一行拜访不下会另起一行，面板默认使用的是流式布局

图形用户界面, 文本, 应用程序

描述已自动生成

1. 表格布局

文本

描述已自动生成

# 事件

在GUI中指的是点击了某个组件，然后执行指定的方法

实现事件的步骤

1. 定义一个类实现ActionListener接口

文本

描述已自动生成

1. 给按钮添加事件

表格

描述已自动生成

出现的问题？无法区分点击了哪个按钮，如果给每个按钮都写一个事件，比较麻烦

通过事件的参数来获取事件源（哪个按钮被点击了）

手机屏幕截图

描述已自动生成

新问题：如何比较？通过内部类来解决以上问题

图形用户界面, 文本, 应用程序

描述已自动生成

图形用户界面, 应用程序, Word

描述已自动生成

就近原则：优先使用自身类上的属性，如果没有找到，则在外部类中寻找

图形用户界面, 文本, 应用程序

描述已自动生成

如果内部类属性和外部类属性同名，非要使用外部类属性可以使用外部类.this.属性

文本

描述已自动生成

# 如何比较两个按钮

public class MyFrame4 extends JFrame {

private JPanel jPanel;

private JButton jButton1,jButton2,jButton3,jButton4,jButton5,jButton6;

public MyFrame4(){

jPanel = new JPanel();

//可以通过setLayout设置布局

//GridLayout表格布局，然后要指定rows(行数),cols(列数)

jPanel.setLayout(new GridLayout(3,2));

jButton1 = new JButton("东");jButton2 = new JButton("南");

jButton3 = new JButton("西");jButton4 = new JButton("北");

jButton5 = new JButton("中");jButton6 = new JButton("发");

//给所有按钮添加事件

jButton1.addActionListener(new Event1());

jButton2.addActionListener(new Event1());

jButton3.addActionListener(new Event1());

jButton4.addActionListener(new Event1());

jButton5.addActionListener(new Event1());

jButton6.addActionListener(new Event1());

jPanel.add(jButton1);jPanel.add(jButton2);jPanel.add(jButton3);

jPanel.add(jButton4);jPanel.add(jButton5);jPanel.add(jButton6);

this.add(jPanel);

this.setTitle("测试菜单");

this.setSize(300,500);

this.setLocationRelativeTo(null);

this.setVisible(true);

this.setDefaultCloseOperation(EXIT\_ON\_CLOSE);

}

public class Event1 implements ActionListener {

@Override

public void actionPerformed(ActionEvent e) {

//通过e.getSource获取事件源（该事件由哪个组件触发的）

if(e.getSource() == jButton1) {

System.out.println("东被点击了");

} else if(e.getSource() == jButton2) {

System.out.println("南被点击了");

} else if(e.getSource() == jButton3) {

System.out.println("西被点击了");

}else if(e.getSource() == jButton4) {

System.out.println("北被点击了");

}else if(e.getSource() == jButton5) {

System.out.println("中被点击了");

}else if(e.getSource() == jButton6) {

System.out.println("发被点击了");

}

}

}

}

课后练习：给菜单项添加事件，区分哪个菜单项被点击了