学号 E21514033 专业 软件工程 姓名 何铭春

实验日期 **5.31**  教师签字 成绩

实验报告

【实验名称】 Java实验七

【实验目的】

1. 熟悉Java图形布局
2. 能够组合使用Java swing函数

【实验原理】

1. FlowLayout 布局能够依次排列组件，并指定对齐方式。
2. BorderLayout 布局能够指定组件出现的位置
3. CardLayout 能够形成类似扑克牌的显示效果
4. BoxLayout 能够自由组合组件位置
5. Menubar中通过嵌套可以实现复杂的菜单。

【实验内容】

1. 通用frame类

class MyJFrame extends JFrame {

MyJFrame(String s,int x,int y,int w,int h){

setTitle(s);

setBounds(x,y,w,h);

setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);

}

}

1. FlowLayout 布局

public class Layout1 {

public static void main(String[] args){

MyJFrame myJFrame = new MyJFrame("我的窗口",100,100,300,100);

myJFrame.setLayout(new FlowLayout());

JButton button1 = new JButton("北");

myJFrame.add(button1);

JButton button2 = new JButton("东");

myJFrame.add(button2);

JButton button3 = new JButton("西");

myJFrame.add(button3);

JButton button4 = new JButton("南");

myJFrame.add(button4);

JButton button5 = new JButton("中");

myJFrame.add(button5);

myJFrame.pack();

myJFrame.setVisible(true);

}

}

实验截图：



1. BorderLayout 布局

public class Layout2 {

public static void main(String[] args) {

MyJFrame myJFrame = new MyJFrame("我的窗口", 100, 100, 300, 300);

myJFrame.setLayout(new BorderLayout());

JButton button1 = new JButton("东");

JButton button2 = new JButton("南");

JButton button3 = new JButton("西");

JButton button4 = new JButton("北");

JButton button5 = new JButton("中");

myJFrame.add(button1,BorderLayout.EAST);

myJFrame.add(button2,BorderLayout.SOUTH);

myJFrame.add(button3,BorderLayout.WEST);

myJFrame.add(button4,BorderLayout.NORTH);

myJFrame.add(button5,BorderLayout.CENTER);

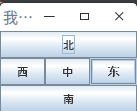
myJFrame.pack();

myJFrame.setVisible(true);

}

}

实验截图：



1. 嵌套菜单

public class Layout3 {

public static void main(String[] args){

MyJFrame2 myJFrame2 = new MyJFrame2("我的窗口",100,100,300,150);

myJFrame2.setVisible(true);

}

}

class MyJFrame2 extends JFrame {

JMenuBar menuBar;

JMenu menu1,menu2,submenu;

JMenuItem item1,item2,item3,item4;

MyJFrame2(String s,int x,int y,int w,int h){

setMenu();

setLayout(new FlowLayout());

setTitle(s);

setBounds(x,y,x,h);

setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);

}

void setMenu(){

menuBar = new JMenuBar();

menu1 = new JMenu("File");

menu2 = new JMenu("Edit");

item1 = new JMenuItem("open");

item2 = new JMenuItem("close");

item3 = new JMenuItem("exit");

menu1.add(item1);

menu1.add(item2);

menu1.addSeparator();

menu1.add(item3);

item1 = new JMenuItem("cut");

item2 = new JMenuItem("copy");

item3 = new JMenuItem("paste");

menu2.add(item1);

menu2.add(item2);

menu2.add(item3);

menu2.addSeparator();

submenu = new JMenu("find and replace");

item1 = new JMenuItem("find");

item2 = new JMenuItem("replace");

submenu.add(item1);

submenu.add(item2);

menu2.add(submenu);

menuBar.add(menu1);

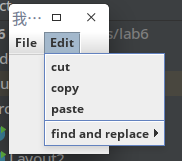
menuBar.add(menu2);

setJMenuBar(menuBar);

}

}

实验截图：



1. 嵌套布局

public class Layout4 {

public static void main(String[] args) {

MyJFrame myJFrame = new MyJFrame("我的窗口",100,100,300,500);

myJFrame.setLayout(new FlowLayout());

JPanel panel1 = new JPanel();

panel1.setBackground(Color.RED);

panel1.setPreferredSize(new Dimension(300,100));

JPanel panel2 = new JPanel();

panel2.setBackground(Color.ORANGE);

panel2.setPreferredSize(new Dimension(300,100));

JPanel panel3 = new JPanel();

panel3.setBackground(Color.yellow);

panel3.setPreferredSize(new Dimension(300,100));

JPanel panel4 = new JPanel();

panel4.setBackground(Color.GREEN);

panel4.setPreferredSize(new Dimension(300,100));

JPanel panel5 = new JPanel();

panel5.setBackground(Color.blue);

panel5.setPreferredSize(new Dimension(300,100));

Box baseBox = Box.createVerticalBox();

baseBox.add(panel1);

baseBox.add(panel2);

baseBox.add(panel3);

baseBox.add(panel4);

baseBox.add(panel5);

myJFrame.add(baseBox);

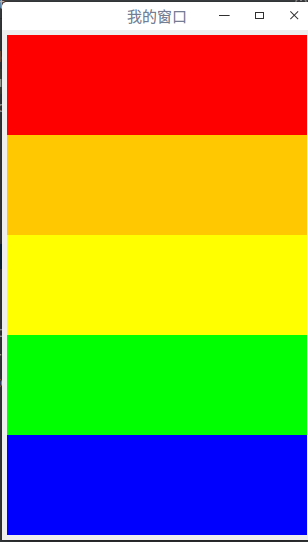
myJFrame.pack();

myJFrame.setVisible(true);

}

}

实验截图：



1. CardLayout 布局

public class Layout5 {

public static void main(String[] args){

MyJFrame myJFrame = new MyJFrame("我的窗口",100,100,500,500);

JPanel panel1 = new JPanel();

CardLayout card = new CardLayout();

panel1.setLayout(card);

panel1.setBackground(Color.yellow);

JPanel panel2 = new JPanel();

myJFrame.add(panel1,BorderLayout.CENTER);

myJFrame.add(panel2,BorderLayout.SOUTH);

String[] names = {"第一张","第二张","第三张","第四张","第五张"};

JLabel label;

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

label = new JLabel(names[i],SwingConstants.CENTER);

label.setFont(new Font("宋体",Font.PLAIN,28));

panel1.add(names[i],label);

}

JButton button1 = new JButton("上一张");

button1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

card.previous(panel1);

}

});

JButton button2 = new JButton("下一张");

button2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

card.next(panel1);

}

});

JButton button3 = new JButton("第一张");

button3.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

card.first(panel1);

}

});

JButton button4 = new JButton("最后一张");

button4.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

card.last(panel1);

}

});

panel2.add(button1);

panel2.add(button2);

panel2.add(button3);

panel2.add(button4);

myJFrame.setVisible(true);

}

}

实验截图：



【小结或讨论】

1. swing图形界面的本质是层层叠加
2. 通过布局能够调整显示效果

3.DardLayout 中已经封装了切换函数