

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM
Khoa Công Nghệ Thông Tin
--❧--

BÀI TẬP MÔN LẬP TRÌNH JAVA

(JAVA PROGRAMMING EXERCISES)

Năm 2014

MODULE 0

(Thực hành: 0 tiết, Tự học: 15 tiết)

Chương I. Giới thiệu về Lập trình hướng đối tượng

Chương II. Các khái niệm cơ sở trong java

Mục tiêu:

- *Hiểu rõ các đặc điểm, môi trường phát triển - hoạt động, khả năng ứng dụng của ngôn ngữ lập trình hướng đối tượng cụ thể (Java).*

Bài tập: Ôn tập các kiến thức về lập trình hướng đối tượng với java (*file bài tập lập trình hướng đối tượng với java*)

MODULE 01, 02, 03 & 04

(Thực hành: 12 tiết, Tự học: 27 tiết)

Chương III. GUI – IO TRONG JAVA

Mục tiêu:

- Hiểu rõ được các khái niệm Containers, Components, Layout Manager, Events.
- Vận dụng thành thạo các đối tượng giao diện chứa (containers) các đối tượng khác trong ngôn ngữ lập trình Java (JFrame, JApplet, JPanel)
- Vận dụng thành thạo các đối tượng giao diện trong giao diện của ứng dụng (JLabel, JButton, JCheckBox, JRadioButton, JPasswordField, JTextArea, JPasswordField)
- Hiểu rõ được khái niệm kiểm soát sự kiện (event handling) trong ngôn ngữ lập trình Java.
- Nắm được mô hình ủy thác sự kiện (event delegation model) và các loại sự kiện (event) trong ngôn ngữ lập trình Java.
- Sử dụng thành thạo các thành phần giao diện (JScrollPane, JSlider, JProgressBar, JFormattedTextField, JEditorPane) các đối tượng hộp thoại (JOptionPane, InputDialog, OptionDialog, JDialog) trong ngôn ngữ lập trình Java.
- Sử dụng thành thạo các đối tượng thiết kế thực đơn (menu) cho giao diện ứng dụng (JMenuBar, JMenu, JMenuItem, JCheckBoxMenuItem, JRadioButtonMenuItem, JPopupMenu, JToolBar).
- Vận dụng thành thạo các khái niệm về luồng nhập xuất và sự tuần tự hóa đối tượng (serializable) trong ngôn ngữ lập trình Java.

Bài Tập 1. Thực hành containers – Hiện thị JFrame

```
package demo.gui;
import javax.swing.JFrame;
public class DemoJFrame extends JFrame{
    public DemoJFrame() {
        setTitle("Demo JFrame"); //Tiêu đề JFrame
        setSize(300, 200); //Kích thước của JFrame
        setDefaultCloseOperation(EXIT_ON_CLOSE); //Thoát chương trình khi click nút exit
        setLocationRelativeTo(null); //Canh giữa màn hình
        setResizable(false); //không cho phép thay đổi kích thước JFrame
        //các thuộc tính khác JFrame
    }
    public static void main(String[] args) {
        new DemoJFrame().setVisible(true); //Hiện thị JFrame
    }
}
```

Bài Tập 2. Thực hành containers – Hiện thị JDialogs

```
package demo.gui;
import javax.swing.JDialog;
public class DemoJDialog extends JDialog{
    public DemoJDialog() {
        setTitle("Demo JDialog"); //Tiêu đề của hộp thoại JDialog
        setDefaultCloseOperation(DISPOSE_ON_CLOSE); //Đóng hộp thoại
        setSize(300, 200);
        setResizable(false);
        //Các thuộc tính khác
    }
}
```

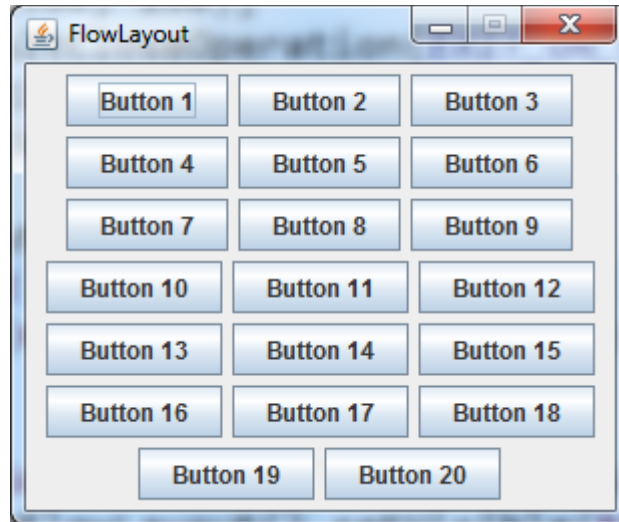
```

    public static void main(String[] args) {
        new DemoJDialog().setVisible(true);
    }
}

```

Bài Tập 3. Thực hành containers – Hiện thị JWindow

Bài Tập 4. Thực hành Layout Managers - FlowLayout



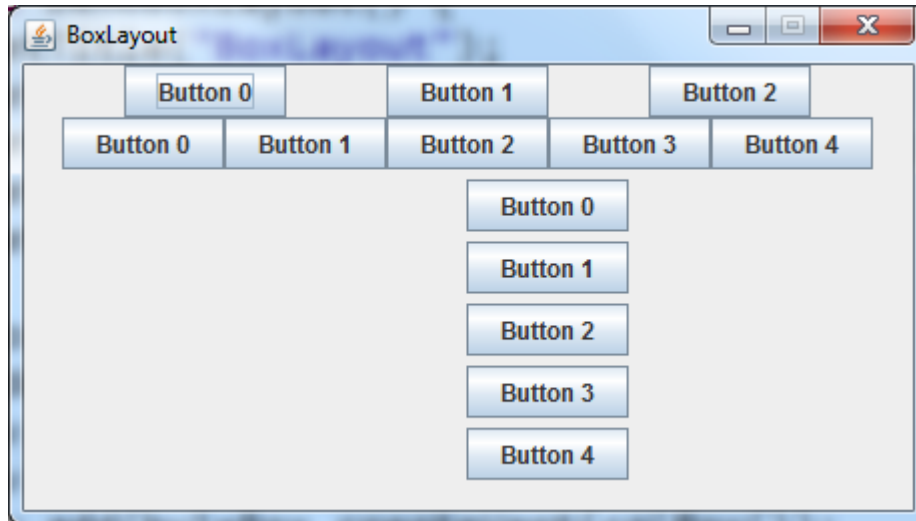
```

package demo.gui;
import java.awt.FlowLayout;
import javax.swing.JButton;
import javax.swing.JFrame;
public class DemoFlowLayout extends JFrame{
    public DemoFlowLayout() {
        setTitle("FlowLayout");
        setSize(300, 250);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        setResizable(false);

        setLayout(new FlowLayout());
        for (int i = 1; i <=20; i++) {
            add(new JButton("Button "+i));
        }
    }
    public static void main(String[] args) {
        new DemoFlowLayout().setVisible(true);
    }
}

```

Bài Tập 5. Thực hành Layout Managers - BoxLayout



```
package demo.gui;
import javax.swing.*;
public class DemoBoxLayout extends JFrame{
    public DemoBoxLayout() {
        setTitle("BoxLayout");
        setSize(450, 250);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        setResizable(false);

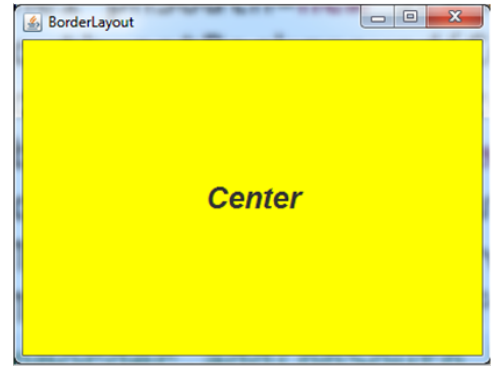
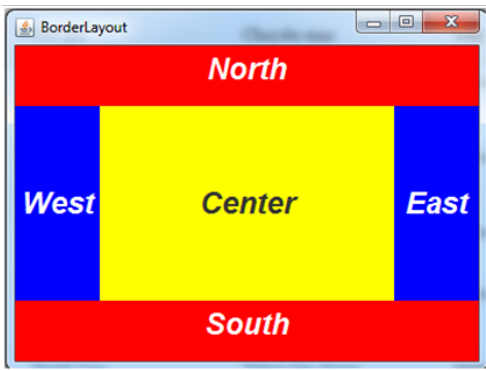
        Box bv,bh1,bh2,bv1;
        add(bv=Box.createVerticalBox());
        bv.add(bh1=Box.createHorizontalBox());
        bv.add(bh2=Box.createHorizontalBox());
        bv.add(bv1=Box.createVerticalBox());

        for(int i=0;i<3;i++){
            bh1.add(Box.createHorizontalGlue());
            bh1.add(new JButton("Button "+i));
        }
        bh1.add(Box.createHorizontalGlue());

        for (int i = 0; i < 5; i++) {
            bh2.add(new JButton("Button "+i));
        }

        for (int i = 0; i < 5; i++) {
            bv1.add(Box.createVerticalStrut(5));
            bv1.add(new JButton("Button "+i));
        }
        bv1.add(Box.createVerticalStrut(5));
    }
    public static void main(String[] args) {
        new DemoBoxLayout().setVisible(true);
    }
}
```

Bài Tập 6. Thực hành Layout Managers - BorderLayout.



```
package demo.gui;

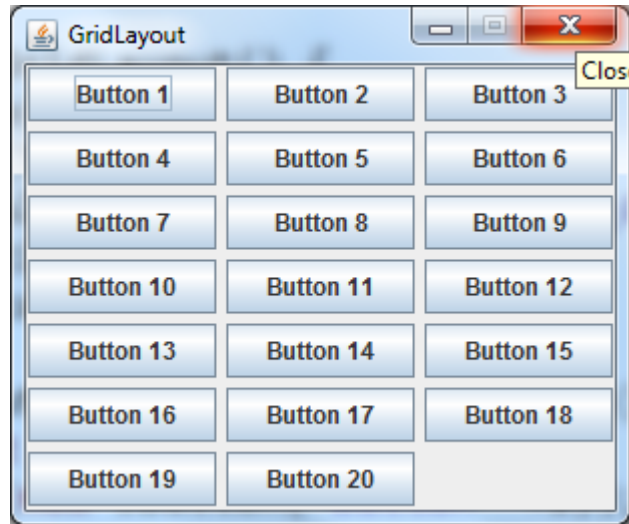
import java.awt.BorderLayout;
import java.awt.Color;
import javax.swing.JButton;
import javax.swing.JFrame;
public class DemoBorderLayout extends JFrame{
    private JButton
        bn=new JButton("North"),
        bs=new JButton("South"),
        be=new JButton("East"),
        bw=new JButton("West"),
        bc=new JButton("Center");

    public DemoBorderLayout() {
        setTitle("BorderLayout");
        setSize(300, 200);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        setResizable(false);

        add(BorderLayout.NORTH, bn);
        add(BorderLayout.SOUTH, bs);
        add(BorderLayout.EAST, be);
        add(BorderLayout.WEST, bw);
        add(BorderLayout.CENTER, bc);

        bn.setBackground(Color.red);
        bn.setForeground(Color.white);
        be.setBackground(Color.blue);
        be.setForeground(Color.white);
        //etc.
    }
    public static void main(String[] args) {
        new DemoBorderLayout().setVisible(true);
    }
}
```

Bài Tập 7. Thực hành Layout Managers - GridLayout



```
package demo.gui;
import java.awt.GridLayout;
import javax.swing.JButton;
import javax.swing.JFrame;
public class DemoGridLayout extends JFrame{
    public DemoGridLayout() {
        setTitle("GridLayout");
        setSize(300, 250);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLocationRelativeTo(null);
        setResizable(false);

        setLayout(new GridLayout(7,3,5,5)); //row=7, column=3, h=5, v=5
        for (int i = 1; i <=20; i++) {
            add(new JButton("Button "+i));
        }
    }
    public static void main(String[] args) {
        new DemoGridLayout().setVisible(true);
    }
}
```

Bài Tập 8. Thực hành Layout Managers – Thiết kế giao diện sau:

