

TEHNICI AVANSATE DE PROGRAMARE

LABORATORUL 5

1. Modelul AWT:

Toate obiectele grafice sunt extinse din clasa **java.awt.Component**. Crearea unui obiect grafic presupune așezarea acestuia pe o suprafață de afișare (fereastră sau applet).

Suprafețele pe care sunt plasate componentele sunt extinse din clasa **java.awt.Container**.

Pentru poziționarea și dimensionarea obiectelor grafice este necesar un gestionar de poziționare.

Pentru tratarea evenimentelor este necesar un ascultător de evenimente.

Ierarhia AWT:

```
Component  TextComponent  TextArea
                                TextField
        Button
        Canvas
        Checkbox
        Choice
        Container  Panel      Applet
                   Window     Frame
                   ScrollPane Dialog
        Label
        List
        Scrollbar
```

2. Gestionari de poziționare:

Poziționare absolută: necesită specificare dimensiunilor și poziției componentelor.

EXP 1:

```
import java.awt.Button;
import java.awt.Frame;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;
public class TestDefaultLayout{
    public static void main(String args[]){
        Frame fer= new Frame("Default Layout");
        fer.addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        });
        fer.setSize(250, 200);
        Button a = new Button("buton A");
        Button b = new Button("buton B");
        a.setBounds(20, 50, 100, 20);
        b.setBounds(130, 50, 100, 20);
        fer.setLayout(null);
        fer.add(a);
        fer.add(b);
        fer.setVisible(true);
    }
}
```



EXE 1: Modificați exemplul anterior astfel încât fereastra să ocupe un sfert suprafața ecranului.

Indicație Folosiți clasa `java.awt.Toolkit` și metoda `getToolkit()` a clasei `Frame`.



EXE 2: Care este gestionarul de poziționare implicit pentru `Frame`?

Gestionarul `FlowLayout`

EXP 2:

```
import java.awt.*;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;
public class TestFlowLayout {
    public static void main(String args[]){
        Frame fer = new Frame("Flow Layout");
        fer.addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        });
        fer.setBounds(50, 50, 500, 200);
        Label l = new Label("eticheta");
        List lst = new List(3, true);
        lst.add("optiunea 1", 1);
        lst.add("optiunea 2", 2);
        lst.add("optiunea 3", 3);
        lst.add("optiunea 4", 4);
        Button b = new Button("buton");
        Panel p = new Panel();
        p.setBackground(new Color(150,0,0));
        Checkbox cb = new Checkbox("optiune");
        Choice c = new Choice();
        c.add("optiune 1");
        c.add("optiune 2");
        p.add(cb);
        p.add(c);
        TextField t = new TextField(10);
        fer.setLayout(new FlowLayout(FlowLayout.RIGHT));
        fer.add(l);
        fer.add(lst);
        fer.add(b);
        fer.add(p);
        fer.add(t);
    }
}
```

Gestionarul `BorderLayout`

EXP 3: Observați cum se redimensionează componentele atunci când se redimensionează fereastra.

```
import java.awt.*;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;
public class TestBorderLayout {
    public static void main ( String args []) {
        Frame fer = new Frame (" Border Layout ");
        fer.addWindowListener(new WindowAdapter(){
```

```

        public void windowClosing(WindowEvent e) {
            System.exit(0);
        }
    });
    fer.setLayout (new BorderLayout ());
    fer.add(new Button (" Nord "), BorderLayout . NORTH );
    fer.add(new Button (" Sud"), BorderLayout . SOUTH );
    fer.add(new Button (" Est"), BorderLayout . EAST );
    fer.add(new Button (" Vest "), BorderLayout . WEST );
    fer.add(new Button (" Centru "), BorderLayout . CENTER );
    fer.pack ();
    fer.setVisible(true);
}
}

```

Gestionarul GridLayout

EXP 4:

```

import java.awt.*;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;
public class TestGridLayout {
    public static void main ( String args []) {
        Frame fer = new Frame (" Grid Layout ");
        fer.addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        });
        fer.setLayout (new GridLayout (3, 2));
        fer.add(new Button ("1"));
        fer.add(new Button ("2"));
        fer.add(new Button ("3"));
        fer.add(new Button ("4"));
        fer.add(new Button ("5"));
        fer.add(new Button ("6"));
        fer.pack ();
        fer.setVisible(true);
    }
}

```

Gestionarul CardLayout

La un moment dat, numai o singură componentă este vizibilă (“cea de deasupra”). Metode ale clasei CardLayout:

- next(Container parinte) urmatoarea componentă
- last(Container parinte) ultima componentă
- first(Container parinte) prima componentă

Gestionarul GridBagLayout

EXP 5:

```
import java.awt.*;
import java.awt.event.*;
class Happy extends Canvas{
    Happy(){
        setBackground(Color.YELLOW);
    }
    public Dimension getPreferredSize(){
        return new Dimension(70,70);
    }
    public void paint(Graphics g){
        g.drawOval(10, 10, 50, 50);
        g.fillOval(25, 25, 5, 5);
        g.fillOval(45, 25, 5, 5);
        g.drawArc(18, 25, 35, 30, 0, -180);
    }
}

public class TestGridBagLayout{
public static void main ( String args []) {
    Frame fer = new Frame (" Grid Bag Layout ");
    fer.addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent e) {
            System.exit(0);
        }
    });
    fer.setVisible(true);
    fer.setSize(600,300);
    GridBagLayout gbc = new GridBagLayout();
    fer.setLayout(gbc);
    final Label lNume = new Label("NUME");
    final Label lPrenume = new Label("PRENUME");
    final Label mesaj = new Label();
    final TextField txtNume = new TextField(50);
    final TextField txtPrenume = new TextField(50);
    final Button bIesire = new Button("IESIRE");
    bIesire.addActionListener(new ActionListener(){
    public void actionPerformed(ActionEvent e) {
        System.exit(0);
    }
    });
    Happy h = new Happy();
    h.addMouseListener(new MouseAdapter(){
        public void mouseClicked(MouseEvent e){
            if (txtNume.getText().compareTo("")==0 ||
                txtPrenume.getText().compareTo("")==0){
                mesaj.setText("Informatii insuficiente");
            }
            else
                mesaj.setText("");
        }
    });
    GridBagConstraints gconst = new GridBagConstraints();
    gconst.gridx = 0;
    gconst.gridy = 0;
    gconst.anchor = GridBagConstraints.WEST;
```

```

gbc.setConstraints(lNume, gconst);
gconst.gridx = 0;
gconst.gridy = 1;
gbc.setConstraints(lPrenume, gconst);
gconst.gridx = 1;
gconst.gridy = 0;
gconst.gridwidth = 2;
gbc.setConstraints(txtNume, gconst);
gconst.gridx = 1;
gconst.gridy = 1;
gconst.gridwidth = 2;
gbc.setConstraints(txtPrenume, gconst);
gconst.gridx = 1;
gconst.gridy = 2;
gconst.gridwidth = 1;
gbc.setConstraints(bIesire, gconst);
gconst.gridx = 2;
gconst.gridy = 2;
gconst.gridwidth = 1;
gconst.fill=GridBagConstraints.BOTH;
gbc.setConstraints(mesaj, gconst);
gconst.gridx = 3;
gconst.gridy = 0;
gconst.gridheight = 2;
gbc.setConstraints(h, gconst);
fer.add(lNume);
fer.add(lPrenume);
fer.add(txtNume);
fer.add(txtPrenume);
fer.add(bIesire);
fer.add(mesaj);
fer.add(h);
fer.pack();
}

```

```

}

```