

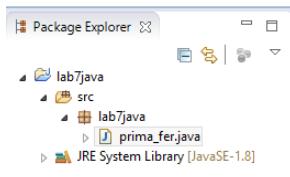
LUCRARE PRACTICĂ NR. 7

Programarea Java

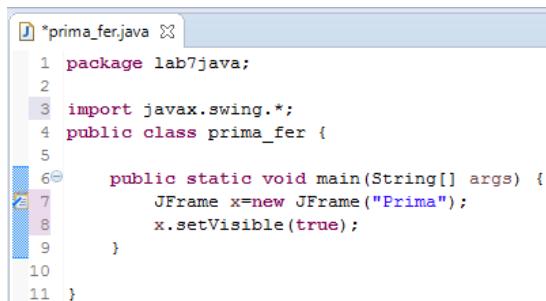
Inițiere în programarea vizuală

- Îndepliniți următoarele sarcini într-un IDE. În continuare se vor ilustra pașii în IDE Eclipse.

1. Creați un nou proiect, cu numele **lab7java**. În acest proiect creaază o clasă **prima_fer**.



2. O fereastră este un obiect al clasei **JFrame** care se află în pachetul **javax.swing**. Pentru a vedea o fereastră este nevoie de a indica faptul că la lansare fereastra va fi vizibilă cu ajutorul metodei **setVisible** și cu argumentul **true**. Codul va arăta astfel:

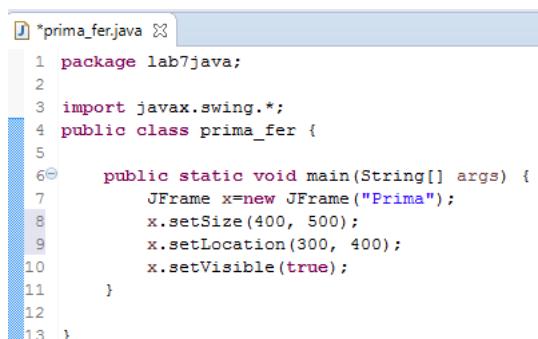


```
1 package lab7java;
2
3 import javax.swing.*;
4 public class prima_fer {
5
6     public static void main(String[] args) {
7         JFrame x=new JFrame("Prima");
8         x.setVisible(true);
9     }
10 }
11 }
```

În rezultat, la lansare fereastra va arăta în felul următor:

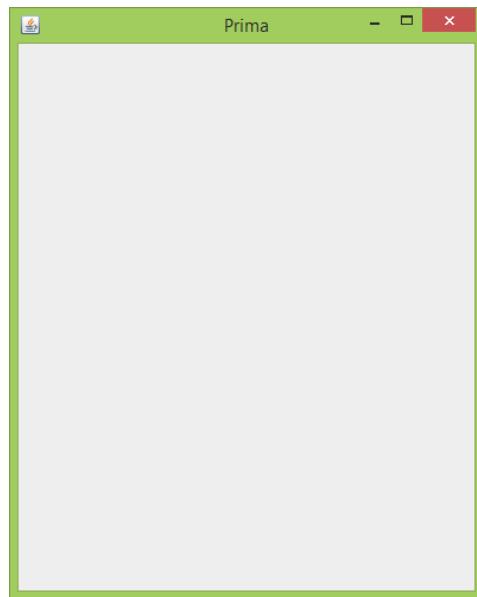


3. Pentru a vedea o formă mai obișnuită este cazul să fie indicate dimensiunile ferestrei cu ajutorul metodei **setSize**. Tot în acest context se folosește metoda **setLocation** pentru a indica locul unde se va lansa fereastra. În condițiile în care aceste metode au fost folosite cu argumentii corespunzători, am obținut următorul program:



```
1 package lab7java;
2
3 import javax.swing.*;
4 public class prima_fer {
5
6     public static void main(String[] args) {
7         JFrame x=new JFrame("Prima");
8         x.setSize(400, 500);
9         x.setLocation(300, 400);
10        x.setVisible(true);
11    }
12 }
13 }
```

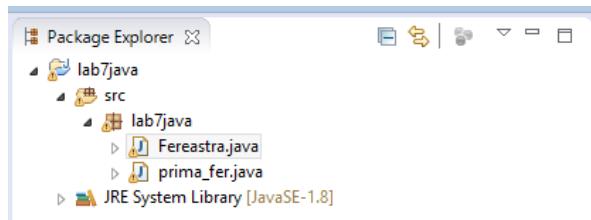
La lansare se va obține următoarea fereastră:



Se observă că în cazul în care dorim să modificăm dimensiunea ferestrei, atunci există posibilitate. Pentru a interzice modificarea se va folosi metoda **setResizable**, cu argumentul **false**.

4. Observăm faptul că codul programului cu mai multe metode devine mai mare. În aceste condiții este mai clar de a separa lucrul cu ferestrele într-o clasă separată. Pentru aceasta se crează o nouă clasă, care o numim **Fereastra** și includem constructorul cu metodele necesare. Codul clasei **Fereastra** va arăta astfel:

```
1 package lab7java;
2
3 import javax.swing.*;
4 public class Fereastra extends JFrame {
5     Fereastra (String nume, int l, int L, int dr, int st){
6         super(nume);
7         setSize(l,L);
8         setLocation(dr,st);
9         setVisible(true);
10    }
11 }
```



5. Pentru a folosi această clasă în metoda **main**, se folosește doar procesul de declarare a obiectului.

```

1 package lab7java;
2
3 public class prima_fer {
4
5     public static void main(String[] args) {
6         Fereastra x=new Fereastra("Prima", 200,300,200,300);
7     }
8
9 }

```

La lansare în execuție fereastra va arăta în felul următor:



6. În continuare vom modifica clasa **Fereastra** și metoda **main** din clasa **prima_fer** în aşa fel încât fereastra să conțină 2 butoane. Codul clasei **Fereastra** va fi următoare:

```

1 package lab7java;
2
3 import java.awt.*;
4 import javax.swing.*;
5 public class Fereastra extends JFrame {
6     Fereastra (String nume){
7         super(nume);
8         setSize(200,100);
9         setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
10        Container x=getContentPane();
11        x.setLayout(new FlowLayout());
12        JButton A=new JButton("Button 1"); x.add(A);
13        JButton B=new JButton("Button 2"); x.add(B);
14        setVisible(true);
15    }
16 }

```

De menționat că a fost importat pachetul `java.awt.*` care conține toate bibliotecile cu componente. Componentele vizuale sunt plasate într-un container cu ajutorul metodei `getContentPane`. Metoda main este asemănătoare celor precedente:

```

1 package lab7java;
2
3 public class prima_fer {
4     public static void main(String[] args) {
5         Fereastra x=new Fereastra("Prima");
6     }
7 }

```

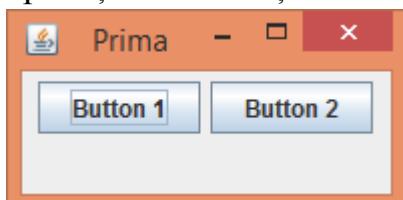
În final fereastra va arăta astfel:



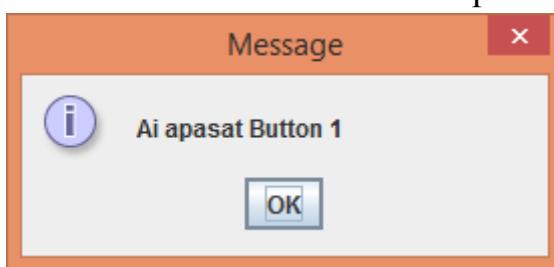
7. Pentru a arăta că aceste butoane pot reacționa, vom adăuga ascultătoare de evenimente (**actionListener**). În acest sens se importează pachetul **java.awt.event.*** și se implementează interfața **ActionListener**. Pentru fiecare buton se adaugă metoda **addActionListener**. Totodată se rescrie metoda **actionPerformed**. În rezultat clasa **Fereastra** va arăta astfel:

```
prima_fer.java  *Fereastra.java X
1 package lab7java;
2
3import java.awt.*;
4import java.awt.event.*;
5import javax.swing.*;
6 public class Fereastra extends JFrame implements ActionListener {
7     Fereastra (String nume){
8         super(nume);
9         setSize(200,100);
10        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
11        Container x=getContentPane();
12        x.setLayout(new FlowLayout());
13        JButton A=new JButton("Button 1"); x.add(A);
14        A.addActionListener(this);
15        JButton B=new JButton("Button 2"); x.add(B);
16        B.addActionListener(this);
17        setVisible(true);
18    }
19
20    public void actionPerformed(ActionEvent ae){
21        if(ae.getActionCommand().compareTo("Button 1")==0)
22            JOptionPane.showMessageDialog(this, "Ai apasat Button 1");
23        else
24            JOptionPane.showMessageDialog(this, "Ai apasat Button 2");
25    }
26 }
```

Metoda **main** din clasa **prima_fer** rămîne neschimbată. La lansarea aplicației se va obține următoarele:



La apăsarea butonului Button 1 va apărea:



Corespunzător la apăsarea Button 2.

ÎN CONTINUARE VOR FI SCRISE CODURILE DE PROGRAM PENTRU ALTE APLICAȚII CARE URMEAZĂ SĂ LE PARCURGETI SAU SĂ LE MODIFICAȚI.

8. Codul programului care creează o fereastră, care inițial ocupă întreg ecranul:

```
import java.awt.*;
import javax.swing.*;
class Fer extends JFrame {
    public Fer(String titlu){
        super(titlu);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Toolkit ec=Toolkit.getDefaultToolkit();
        Dimension dim=ec.getScreenSize();
        int i=dim.height; int l=dim.width;
        setSize(dim.width, dim.height); setVisible(true);
    }
}
public class pv{
    public static void main(String args[]) {
        Fer fp=new Fer("Fereastra pe intreg ecran");
    }
}
```

9. Programul care schimbă culoarea de fondal pe care stau 3 butoane, în momentul în care este apăsat un careva buton.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class Fer extends JFrame implements ActionListener
{
    JPanel panel;
    public Fer(String titlu){
        super(titlu);
        setSize(300,300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container x=getContentPane();
        x.setLayout(new FlowLayout());
```

```

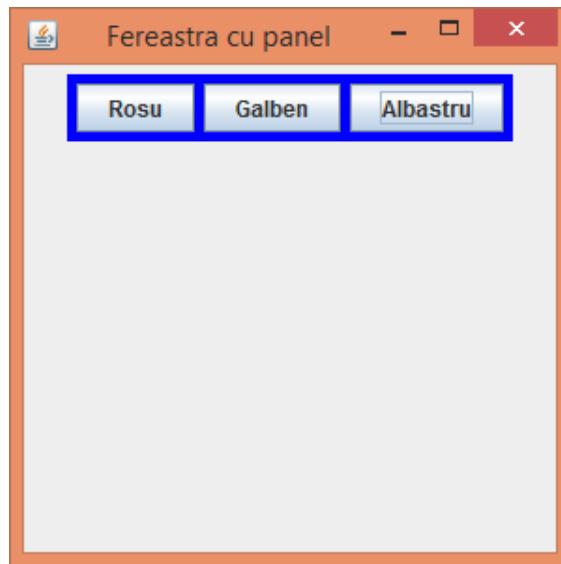
panel=new JPanel();
JButton A=new JButton("Rosu");
panel.add(A); A.addActionListener(this);
JButton B=new JButton("Galben");
panel.add(B); B.addActionListener(this);
JButton C=new JButton("Albastru");
panel.add(C); C.addActionListener(this);
x.add(panel);
setVisible(true);

}

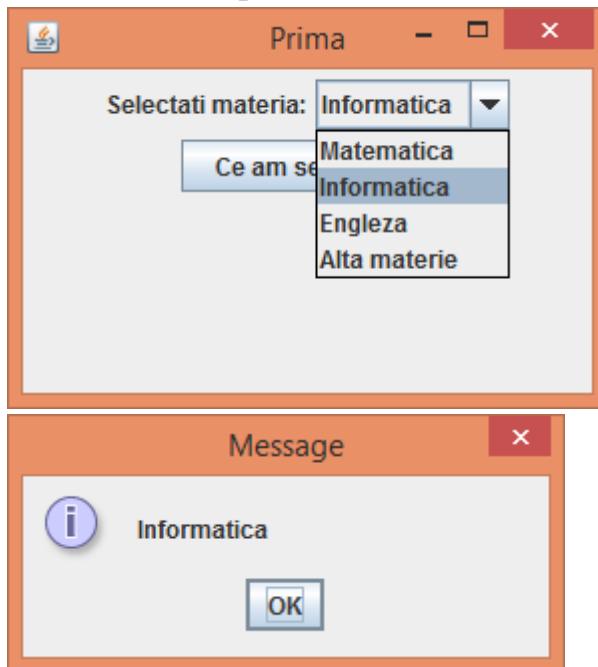
public void actionPerformed (ActionEvent e) {
    String culoare=e.getActionCommand();
    if (culoare.compareTo("Rosu")==0)
        panel.setBackground(Color.red);
    else
        if (culoare.compareTo("Galben")==0)
            panel.setBackground(Color.yellow);
        else
            panel.setBackground(Color.blue);
}
}

public class cv {
    public static void main (String args[]) {
        Fer fp=new Fer("Fereastra cu panel");
    }
}

```



10.O aplicație cu ComboBox, care după selectarea unei discipline și apăsarea butonului corespunzător arată ce a fost selectat, prin afișarea unui mesaj.



```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class Fer extends JFrame implements ActionListener
{
    JComboBox<String> lista;
    public Fer(String titlu){
        super(titlu); setSize(300,200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container x=getContentPane();
        x.setLayout(new FlowLayout());
        JLabel et=new JLabel("Selectati materia:");
        x.add(et);
        lista=new JComboBox<String>();
        lista.addItem("Matematica");
        lista.addItem("Informatica");
        lista.addItem("Engleza");
        lista.addItem("Alta materie");
        lista.setSelectedItem(2);
        x.add(lista);
        JButton but=new JButton("Ce am selectat?");
        x.add(but);
        but.addActionListener(this);
    }
}
```

```

        setVisible(true);
    }
    public void actionPerformed(ActionEvent e){
        JOptionPane.showMessageDialog(this,
    lista.getSelectedItem());      }
}

public class cp{
    public static void main(String args[]){
        Fer fp=new Fer("Exemplu de lista");
    }
}

```

11.Un exemplu de lucru cu CheckBox. Sînt puse mai multe cutii de CheckBox, și unele sînt bifate. La apăsarea butonului în linia de comandă sînt trecute acele discipline care sînt bifate.

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class Fer extends JFrame implements ActionListener
{
    JCheckBox b1,b2,b3,b4;
    public Fer(String titlu){
        super(titlu); setSize(300,200);
        Container x=getContentPane();
        x.setLayout(new FlowLayout(FlowLayout.LEFT));
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JLabel et=new JLabel("Selectati materia:");
        x.add(et);
        b1=new JCheckBox("Matematica");x.add(b1);
        b2=new JCheckBox("Informatica");x.add(b2);
        b3=new JCheckBox("Engleza");x.add(b3);
        b4=new JCheckBox("Alte materie");x.add(b4);
        JButton but=new JButton("Ce am selectat?");
        x.add(but);
        but.addActionListener(this);
        setVisible(true);
    }
    public void actionPerformed(ActionEvent e){

```

```

        if                                (b1.isSelected())
System.out.println(b1.getText());
        if                                (b2.isSelected())
System.out.println(b2.getText());
        if                                (b3.isSelected())
System.out.println(b3.getText());
        if                                (b4.isSelected())
System.out.println(b4.getText());
    }
}

public class pv1{
    public static void main(String args[]){
        Fer fp=new Fer("Exemplu de lista");
    }
}

```

12. Același lucru cu elemente de tip RadioButton. Ca sarcina transformăți afișarea în linia de comandă în cutia de dialog de afișare a mesajului.

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.util.*;

class Fer extends JFrame implements ActionListener
{
    ButtonGroup grup;
    public Fer(String titlu){
        super(titlu); setSize(300,200);
        Container x=getContentPane();
        x.setLayout(new FlowLayout(FlowLayout.LEFT));
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JLabel et=new JLabel("Selectati materia:");
        x.add(et);
        JRadioButton     b1=new     JRadioButton("Matematica");
        x.add(b1);
        JRadioButton     b2=new     JRadioButton("Informatica");
        x.add(b2);
        JRadioButton     b3=new     JRadioButton("Engleza");
        x.add(b3);
        JRadioButton     b4=new     JRadioButton("Alte materie");
        x.add(b4);
        JButton but=new JButton("Ce am selectat?");
        x.add(but);
        but.addActionListener(this);
        grup=new ButtonGroup();
    }
}

```

```

        grup.add(b1);grup.add(b2);grup.add(b3);grup.add(b4);
        setVisible(true);
    }
    public void actionPerformed(ActionEvent e){
        Enumeration enum1=grup.getElements();
        while(enum1.hasMoreElements()){
            JRadioButton
            b=(JRadioButton)enum1.nextElement();
            if(b.isSelected())
                System.out.println(b.getText());
        }
    }
}

public class pv2{
    public static void main(String args[]){
        Fer fp=new Fer("Exemplu de lista");
    }
}

```

PENTRU LUCRU INDIVIDUAL ȘI ANTRENAMENT!

1. Elaborați un program care ar calcula suma a două numere. Fereastra trebuie să aibă 3 elemente de tip Label, TextField și un buton. La apăsarea pe buton în TextField al treilea se înscrive suma a două numere în celelalte elemente de tip TextField.