

LUCRARE PRACTICĂ NR. 11

Programarea Java

Lucrul cu clase înfășurătoare și conversii. Tratarea excepțiilor

- Îndepliniți următoarele sarcini într-un IDE.

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

APLICAȚIA NR.1

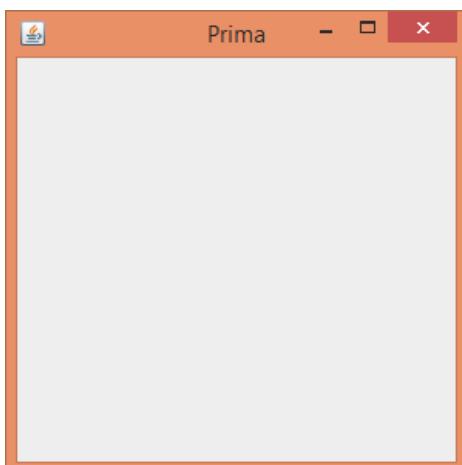
Aplicația va lansa o fereastră unde se vor introduce valori numerice separate prin spații și se va calcula suma acestor numere.

1. Codul programului care creează o fereastră simplă:

```
import java.awt.*;
import javax.swing.*;
public class Fereastra extends JFrame {

    public Fereastra(String titlu){
        super(titlu);
        setSize(300,180);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
        setResizable(false);
    }
}

public class prima_fer {
    public static void main(String[] args) {
        Fereastra x=new Fereastra("Prima");
    }
}
```



2. La formă adăugăm două etichete (**JLabel**), două zone de text (**JTextField**) și un buton (**JButton**). În acest sens clasa fereastra va arăta astfel:

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

class Fereastra extends JFrame {
    String sir;
    JLabel B1;
    JTextField A1;
    int nr1=0,nr2=0;
    public Fereastra (String titlu){
        super(titlu);
        setSize(300,120);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container x=getContentPane();
        x.setLayout(new FlowLayout());
        B1=new JLabel("Lista de numere");
        x.add(B1);
        A1=new JTextField(25);
        x.add(A1);
        JButton A=new JButton("Rezultat");
        x.add(A);
        setVisible(true);
        setResizable(false);
    }
}
```



3. În momentul în care se apasă pe butonul Rezultat, va fi afișată cutia de dialog pentru afișarea mesajului. Mesajul va conține rezultatul sumei numerelor introduse de la în cutiile de text. În acest sens se va adăuga pachetul **java.awt.event.***. La clasa **Fereastra** adăugăm o implementare a interfeței **ActionListener**. În final adăugăm o metodă de prelucrare a evenimentelor **actionPerformed**, în care se va apela caseta de dialog pentru afișarea mesajului corespunzător. În final clasa **Fereastra** va arăta astfel:

```

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

class Fereastra extends JFrame implements ActionListener{
    String sir;
    JLabel B1;
    JTextField A1;
    int nr1=0,nr2=0;
    public Fereastra (String titlu){
        super(titlu);
        setSize(300,120);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container x=getContentPane();
        x.setLayout(new FlowLayout());
        B1=new JLabel("Lista de numere");
        x.add(B1);
        A1=new JTextField(25);
        x.add(A1);
        JButton A=new JButton("Rezultat");
        x.add(A);A.addActionListener(this);
        setVisible(true);
        setResizable(false);
    }

    public void actionPerformed(ActionEvent ae){
        String s=new String(A1.getText());
        StringTokenizer st=new StringTokenizer(s);
        int k=0;
        while(st.hasMoreTokens()) {
            k+=Integer.parseInt(st.nextToken());
        }
        JOptionPane.showMessageDialog(this, "Suma numerelor este
"+k);
    }
}

```

În final programul va funcționa după cum urmează în printScreenuri:

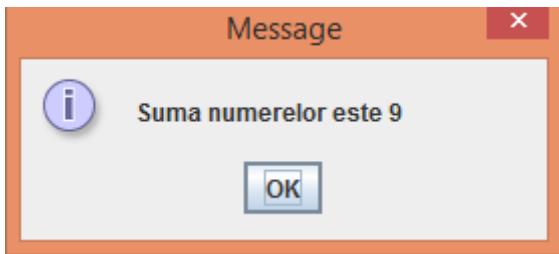
a. Fereastra inițială.



b. Au fost introduse valorile:



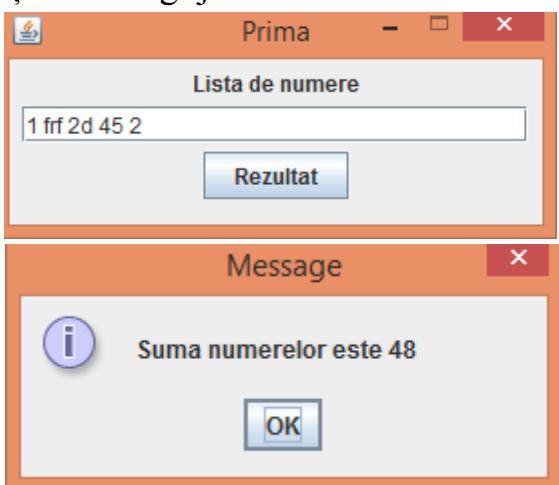
c. La apăsarea butonului Rezultat.



4. Există situația cînd numere nu sînt introduse în format corect. În cazul nostru ar putea fi că textul introdus nu doar un număr întreg. În acest caz trebuie de indicat aceasta. Pentru a rezolva această problemă în metoda **actionPerformed** se adaugă construcțiile **try catch**, pentru tratarea excepțiilor. Metoda completată va arăta astfel:

```
• public void actionPerformed(ActionEvent ae) {
•     String s=new String(A1.getText());
•     StringTokenizer st=new StringTokenizer(s);
•     int k=0;
•
•     while(st.hasMoreTokens()) {
•         try{
•             nr2=Integer.parseInt(st.nextToken());
•             k+=nr2;
•         }
•         catch(NumberFormatException e) { }
•     }
•     JOptionPane.showMessageDialog(this, "Suma numerelor este
• "+k);
• }
```

În acest caz, în momentul în care nu este introdus un număr întreg secvența va fi neglijată:



APLICAȚIA NR. 2

Aplicația va lansa o fereastră unde se vor introduce valori numerice separate prin spații și se va stabili câte numere distințe au fost introduse și care sunt acestea.

1. Repetăm pașii din programul precedent pentru a crea interfața grafică.



2. La prelucrarea datelor introduse metoda **actionPerformed** va arăta astfel:

```
public void actionPerformed(ActionEvent ae) {
    String s=new String(A1.getText());
    StringTokenizer st=new StringTokenizer(s);
    String sr=new String();
    boolean logic;
    Set<Integer> sti=new HashSet();
    while(st.hasMoreTokens()){
        try{
            nr2=Integer.parseInt(st.nextToken());
            logic=sti.add(nr2);
            if (logic)
                sr+=nr2+" ";
        }
        catch(NumberFormatException e) {}
    }
    JOptionPane.showMessageDialog(this, sti.size()+" numere: "+sr);
}
```

- a. Pornind interfața:



b. Introducînd numerele:



c. Afîșarea rezultatului:



De menþionat c  la rezolvare au fost folosite colec iile generice.

LUCRU INDIVIDUAL dup  modelul programelor ilustrate mai sus

În continuare baz ndu-v  pe ultima aplica ie elabora i urm toarele programe:

1. Se citesc  n TextField numerele reale separate prin spa iu. Num rul acestor numere nu este cunoscut. Elabora i un program Java care realizeaz  urm toarele sarcini:
 - a. Dac  nu au fost introdus nici un num r s  apar  un mesaj l amuritor.
 - b. Dac  au fost citite s  se calculeze suma lor.
 - c. C te numere distin te (diferite) au fost introduse. (Indica ii: Se va utiliza colec ia corespun atoare pentru a avea deja numerele diferite)
 - d. Care s nt primele 3 numere maxime citite de la tastatur . (Indica ii: se va utiliza colec ia corespun atoare pentru a avea deja la citire numerele sortate)

Exemplu: Dac  au fost citite numerele: **-4 5 2 5 6 7 -9 1 2 2 -4 7**, atunci trebuie s  fie afi at astfel:

**20
7
7 6 5**