

LUCRARE PRACTICĂ NR. 9

Programarea Java

Lucrul cu sirurile de caractere

- Îndepliniți următoarele sarcini într-un IDE.

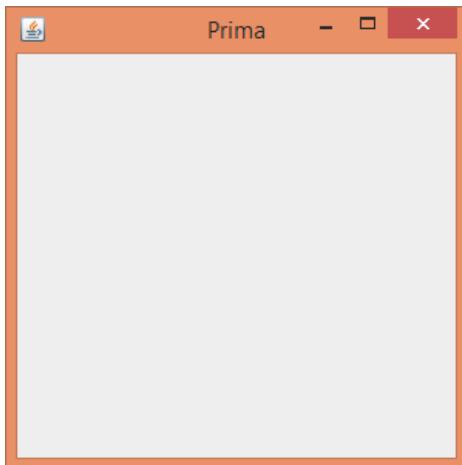
VOR FI SCRISE CODURILE DE PROGRAM PENTRU ALTE APLICAȚII CARE URMEAZĂ SĂ LE PARCURGEȚI SAU SĂ LE MODIFICAȚI.

APLICAȚIA NR.1

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(300,300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }
}

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



2. Programul de mai sus, se mai adaugă un buton, cu textul **Date Noi**. Clasa Fereastra va căpăta următorul cod:

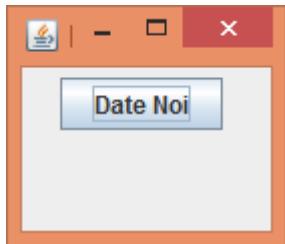
```
import java.awt.*;
import javax.swing.*;
```

```

class Fereastra extends JFrame {
    public Fereastra (String titlu){
        super(titlu);
        setSize(120,120);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container x=getContentPane();
        x.setLayout(new FlowLayout());
        JButton A=new JButton("Date Noi");
        x.add(A);
        setVisible(true);
    }
}

```

Fereastra va arăta astfel:



Observăm că butonul încă nu este unul funcțional.

- Adăugăm pachetul **java.awt.event.***. La clasa fereastra adăugăm o implementare a interfeței **ActionListener**. Adăugăm în constructorul clasei Fereastra un element de tip **JLabel**. În final adăugăm o metodă de prelucrare a evenimentelor **actionPerformed**, în care se va apela caseta de dialog pentru citirea datelor. În final clasa **Fereastra** va arăta astfel:

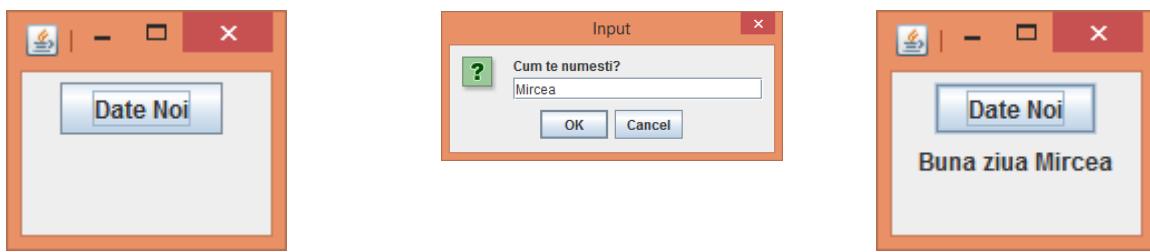
```

import java.awt.event.*;

class Fereastra extends JFrame implements ActionListener {
    String sir;
    JLabel B;
    public Fereastra (String titlu){
        super(titlu);
        setSize(120,120);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container x=getContentPane();
        x.setLayout(new FlowLayout());
        JButton A=new JButton("Date Noi");
        x.add(A);A.addActionListener(this);
        B=new JLabel();
        x.add(B);
        setVisible(true);
    }

    public void actionPerformed (ActionEvent ae){
        sir=JOptionPane.showInputDialog("Cum te numesti?");
        B.setText("Buna ziua "+sir);
    }
}

```



APLICAȚIA NR.2

- Inițial creăm o fereastră cu o etichetă (**JLabel**), o zonă 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;
    public Fereastra (String titlu){
        super(titlu);
        setSize(300,120);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container x=getContentPane();
        x.setLayout(new FlowLayout());
        B1=new JLabel("Dati un sir");
        x.add(B1);
        A1=new JTextField(25);
        x.add(A1);
        JButton A=new JButton("Rezultat");
        x.add(A);
        setVisible(true);
    }
}
  
```



2. În momentul în care se apasă pe butonul Rezultat, va fi afișată cutia de dialog pentru afișarea mesajului. Mesajul va conține lungimea șirului de caractere introdus în cutia textuală. 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.awt.*;
import javax.swing.*;
import java.awt.event.*;

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

    public void actionPerformed (ActionEvent ae){
        int k=A1.getText().length();
        JOptionPane.showMessageDialog(this,"Lungimea sirului
este: "+k);
    }
}
```

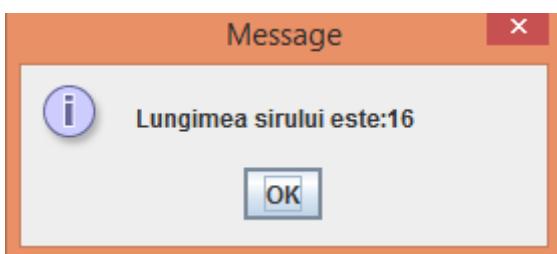
În final programul va funcționa după cum urmează în printScreenuri:
a. Fereastra inițială.



b. A fost introdus textul: Programarea Java



c. La apăsarea butonului Rezultat.



LUCRU INDIVIDUAL după modelul programelor ilustrate mai sus

În continuare bazîndu-vă pe ultima aplicație elaborați următoarele programe:

- a. Caută poziția caracterului într-un sir de caractere. De exemplu, care este caracterul în sir care se află pe poziția k.
- b. Compararea a două siruri de caractere:
 - i. ținînd cont de diferența de litere mari și mici
 - ii. ignorînd diferența între litere mari și mici.
- c. Stabilirea dacă un sir de caractere se începe și se termină cu subșirurile corespunzătoare.
- d. Pe ce poziție se află un subșir în cadrul unui sir, dacă acesta se conține în cel din urmă.
- e. Care este sirul resultant în urma înlocuirii unui subșir prin alt subșir în cadrul unui sir de caractere.
- f. Afîsați numărul de cuvinte într-o frază, în care cuvintele sînt separate prin spațiu liber, semnele de punctuație și semnul -.