

LUCRARE PRACTICĂ NR. 8

Programarea Java

Programarea vizuală

- Îndepliniți următoarele sarcini într-un IDE.

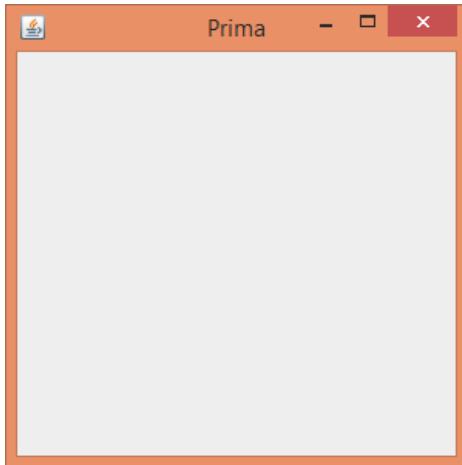
VOR FI SCRISE CODURILE DE PROGRAM PENTRU ALTE APLICAȚII CARE URMEAZĂ SĂ LE PARCURGEȚI SAU SĂ LE MODIFICAȚI.

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");
    }
}
```

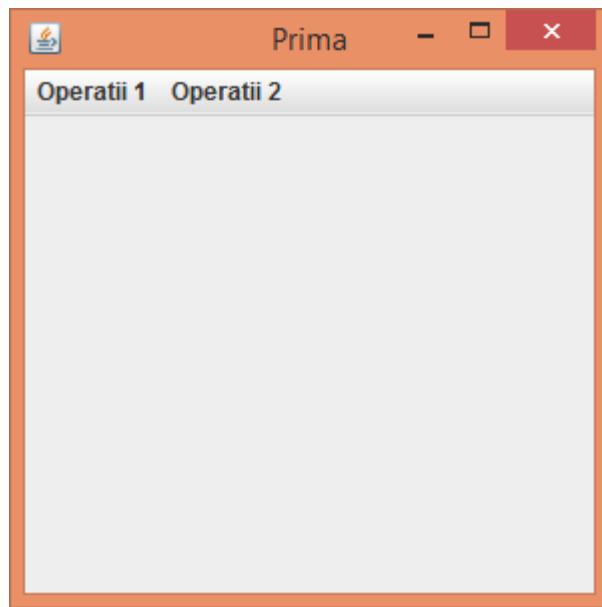


2. Programul de mai sus, se mai adaugă bara de meniu, cu două opțiuni:

```
JMenuBar bara = new JMenuBar();
JMenu meniu1=new JMenu("Operatii 1");
bara.add(meniu1);
JMenu meniu2=new JMenu("Operatii 2");
bara.add(meniu2);
```

```
setJMenuBar (bara) ;
```

Acest text se adaugă în constructorul clasei Fereastra. Dar atent! Textul acesta se adaugă înainte de metoda setVisible. Dacă se va adăuga după, nu se va observa nici un meniu.



Observăm o fereastră cu un meniu cu două opțiuni. Nu este posibil de a face careva manipulări.

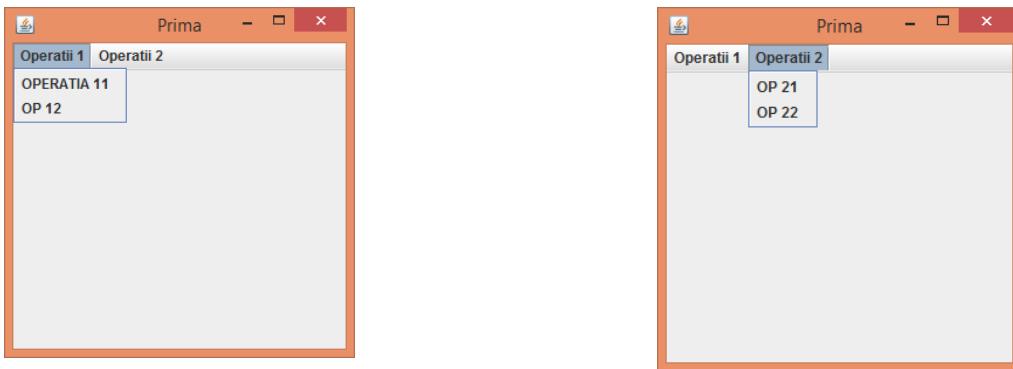
3. Adăugăm pentru prima opțiune – Operații 1 – un meniu cu 2 variante. Pentru asta adăugăm următorul cod:

```
JMenuItem item11=new JMenuItem("OPERATIA 11");  
meniu1.add(item11);  
JMenuItem item12=new JMenuItem("OP 12");  
meniu1.add(item12);
```

Analog și pentru a doua opțiune – Operații 2 – un meniu cu 2 variante. Pentru asta adăugăm următorul cod:

```
JMenuItem item21=new JMenuItem("OP 21");  
meniu2.add(item21);  
JMenuItem item22=new JMenuItem("OP 22");  
meniu2.add(item22);
```

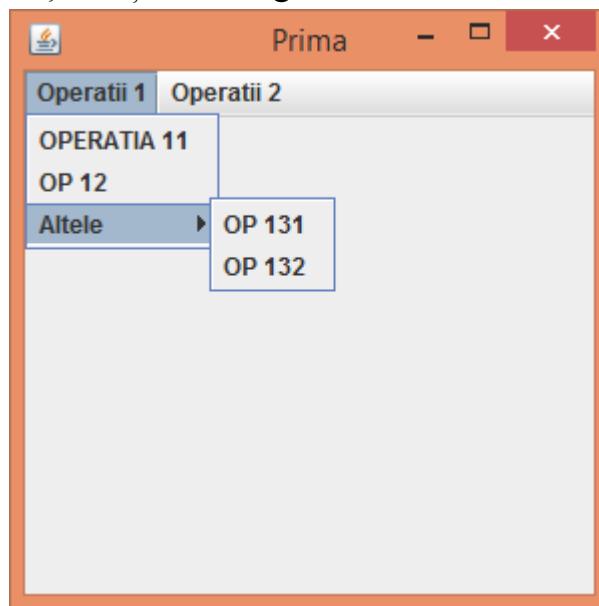
În rezultat meniurile vor arăta după cum urmează:



4. În continuare putem adăuga un submeniu pentru primul meniu. Pentru asta vom scrie codul de mai jos:

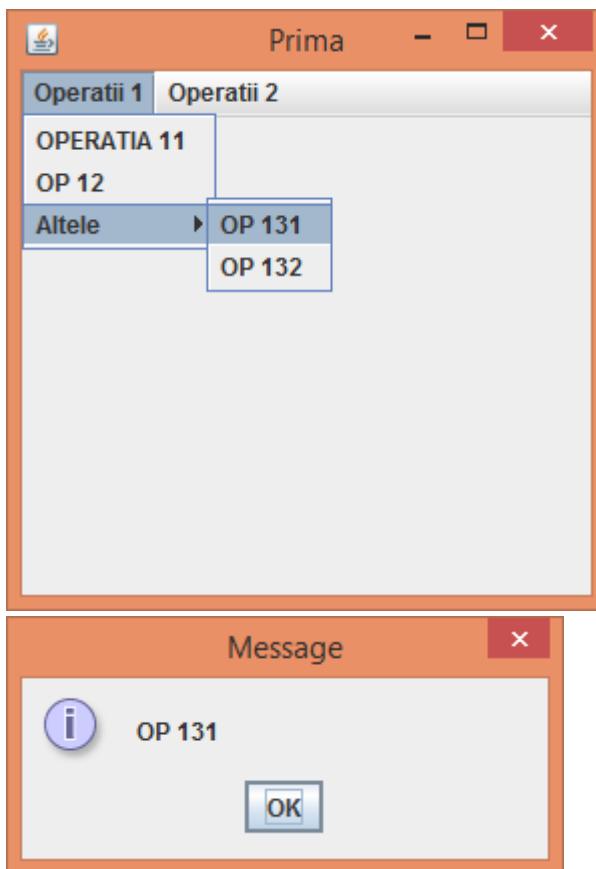
```
JMenu altele=new JMenu("Altele");
menu1.add(altele);
 JMenuItem item131 =new JMenuItem("OP 131");
altele.add(item131);
 JMenuItem item132 =new JMenuItem("OP 132");
altele.add(item132);
```

În rezultat se va obține aşa o configurare a meniului:



5. Pentru fiecare alegere a opțiunii din meniu, se va lansa o cutie de dialog de afișare a mesajului, cu textul opțiunii. Pentru aceasta la început se adaugă importarea pachetului `java.awt.event.*`. Apoi se implementează interfața `ActionListener`. Se adaugă pentru fiecare opțiune din meniu câte un ascultător. Se scrie metoda `actionPerformed`.

În final clasa Fereastra va arăta astfel:



APLICAȚIA NR.2

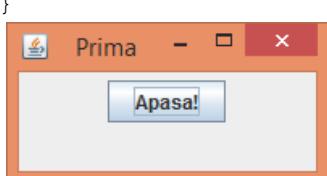
1. Inițial creăm o fereastră cu un buton. În acest sens clasa fereastra va arăta astfel:

```

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

class Fereastra extends JFrame {
    public Fereastra (String titlu){
        super(titlu);
        setSize(200,100);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        Container x=getContentPane();
        x.setLayout(new FlowLayout());
        JButton A=new JButton("Apasa!");
        x.add(A);
        setVisible(true);
    }
}

```



2. În momentul în care se apasă pe butonul din fereastră, trebuie să apară o cutie de dialog cu 3 butoane, care au denumire Buton1, Buton2 și Buton3. În acest sens este nevoie de a adăuga pachetul java.awt.event.*. La clasa Fereastra de adăugat implementarea interfeței ActionListener. La buton de adăuga un ascultător de evenimente cu ajutorul metodei addActionEvent. De scris metoda actionPerformed care va genera apariția cutiei de dialog cu opțiuni.

Metoda actionPerformed va arăta astfel:

```
public void actionPerformed(ActionEvent ae) {
    Object[] butoane={"Buton 1","Buton 2","Buton 3"};
    int n=JOptionPane.showOptionDialog(this, "Apasati un buton",
        "Test showOptionDialog",
        JOptionPane.YES_NO_CANCEL_OPTION,
        JOptionPane.QUESTION_MESSAGE,
        null,
        butoane,
        butoane[2]);
}
```

La apăsarea butonului va apărea cutia de dialog cu opțiuni de genul:



3. Pentru ca să apară o cutie de dialog de afișare a mesajelor cu mesajul corespunzător butonului apăsat, la metoda actionPerformed se va mai adăuga o instrucțiune de comparație care va afla numărul butonului apăsat și în dependență de dinsul va cunoaște care buton a fost apăsat. Numărul butonului este numerotat de la 0 (zero) la 2, de la stînga spre dreapta. Codul care trebuie adăugat la metoda actionPerformed este următoare:

```
if (n==0) JOptionPane.showMessageDialog(this,"butonul 1");
else if (n==1) JOptionPane.showMessageDialog(this,"butonul 2");
else JOptionPane.showMessageDialog(this,"butonul 3");
```

Cutia de dialog corespunzătoare este:

