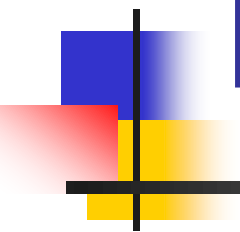


Vartotojo sąsajos kūrimas, naudojant Java Swing biblioteką





Swing komponentų privalumai lyginant su AWT (1)

- Parašyti išimtinai Java kodu ir nepriklauso nuo platformos
- Žymės ir mygtukai be įprasto teksto gali turėti paveikslėlius
- Daugeliui komponentų galima uždėti ar pakeisti rėmelius



Swing komponentų privalumai lyginant su AWT (2)

- Komponentai nebūtinai yra stačiakampės formos (pvz., apvalus mygtukas)
- Lengvai keičiamas komponentų elgesys ar išvaizda, panaudojant esamus metodus ar juos perdengus



Galimos problemos

- Reikia vengti mišraus AWT ir Swing komponentų vartojimo
- Matomų komponentų modifikavimas atliekamas tik iš įvykius apdorojančios gijos (event-dispatching thread)
- Swing komponentus reikėtų dėti tik į Swing viršutinio lygio konteinerius (JFrame, JApplet)



Pagrindiniai Swing paketai

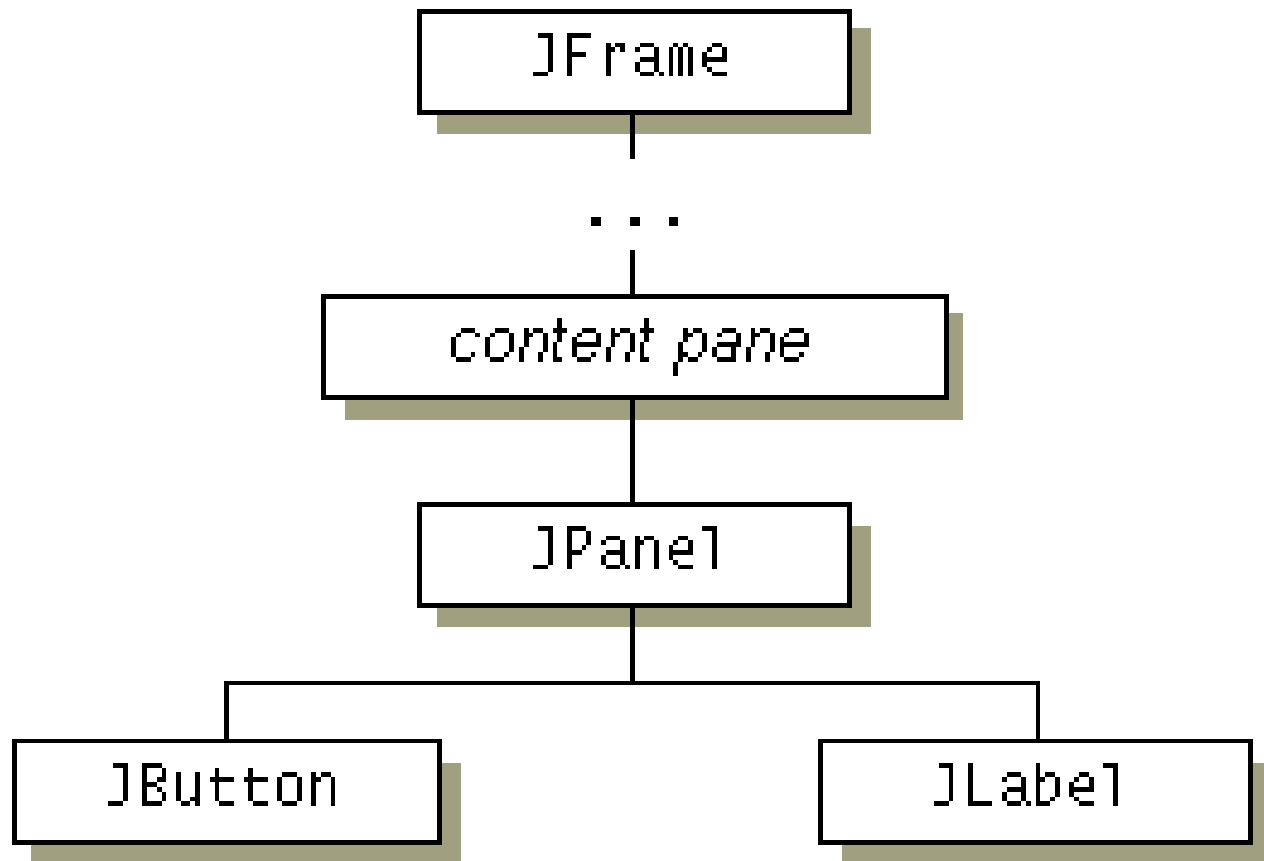
- `import javax.swing.*;`
- `import javax.swing.event.*;`
- `import java.awt.*;`
- `import java.awt.event.*;`



Swing'o komponentai ir jų turinio hierarchija (1)

- Viršutinio lygio konteineriai (JFrame, JApplet, JDialog)
- Tarpiniai konteineriai (JPanel, JScrollPane, JTabbedPane)
- Atominiai (atomic) komponentai, priimančys informaciją ar išvedantys ją vartotojui

Swing'o komponentai ir jų turinio hierarchija (2)





Programos pavyzdys nr.1

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

public class Demo {
    public static void main(String args[]) {
        JFrame frame = new JFrame("Demo");
        frame.addWindowListener(
            new WindowAdapter() {
                public void windowClosing(WindowEvent evt) {
                    System.exit(0);
                }
            }
        );
    }
}

// tęsinys kitoje skaidrėje
```



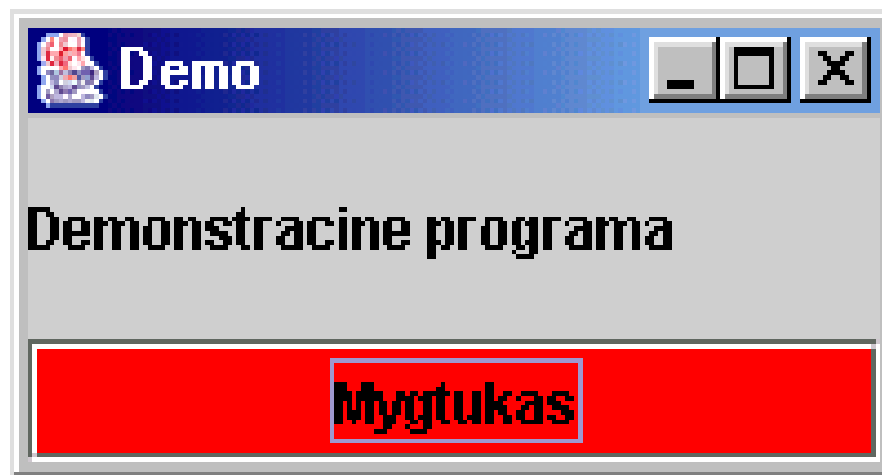

Programos pavyzdys nr.1

```
JLabel label = new JLabel("Demonstracine programa");

JButton button = new JButton("Mygtukas");
button.setBackground(Color.RED);
frame.getContentPane().add(label,
    BorderLayout.CENTER);
frame.getContentPane().add(button,
    BorderLayout.PAGE_END);
frame.setBounds(10, 10, 200, 100);
frame.setVisible(true);
}
}
```



Programos pavyzdys nr.1





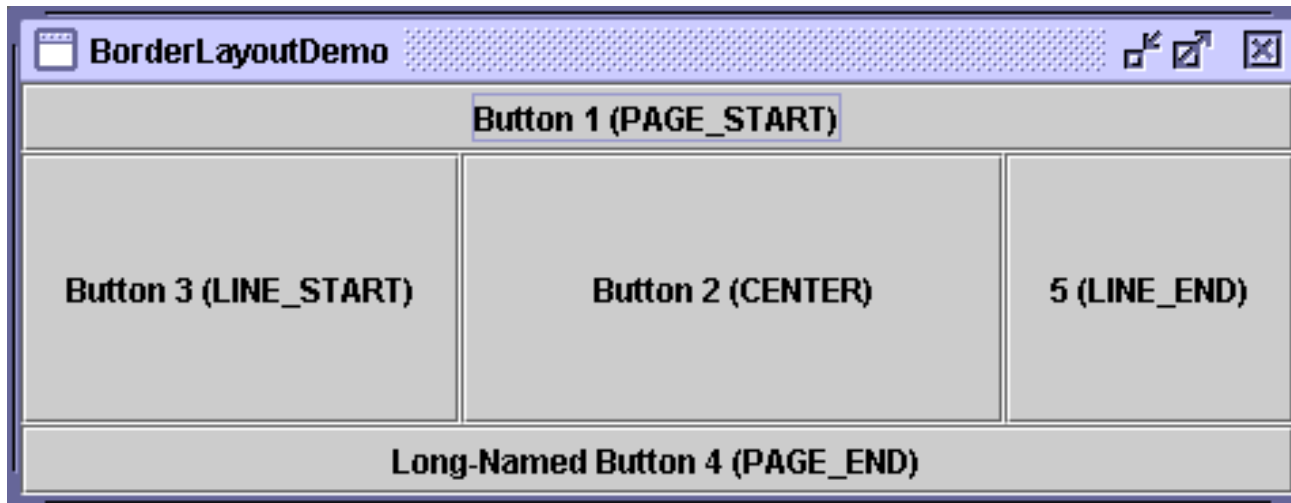
Vartotojo sąsajos komponentų išdėstymo būdai

- Išdėstymo būdas gali būti nustatytas bet kuriam viršutinio lygio konteineriui (pvz., *JFrame*, *JDialog*, *JApplet*, *JPanel* ir *pan.*)
- Išdėstymui nustatyti naudojamas *Container* klasės metodas:

```
public void setLayout(LayoutManager isdestymoBudas)
```

Klasės skirtos nustatyti GUI komponentų išdėstymą (1)

■ BorderLayout



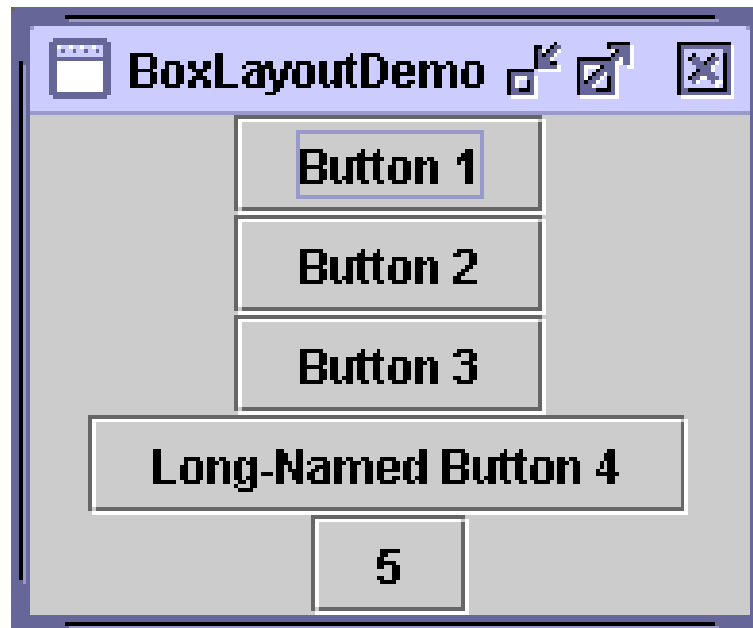
Klasės skirtos nustatyti GUI komponentų išdėstymą (2)

- **FlowLayout**



Klasės skirtos nustatyti GUI komponentų išdėstymą (3)

- BorderLayout



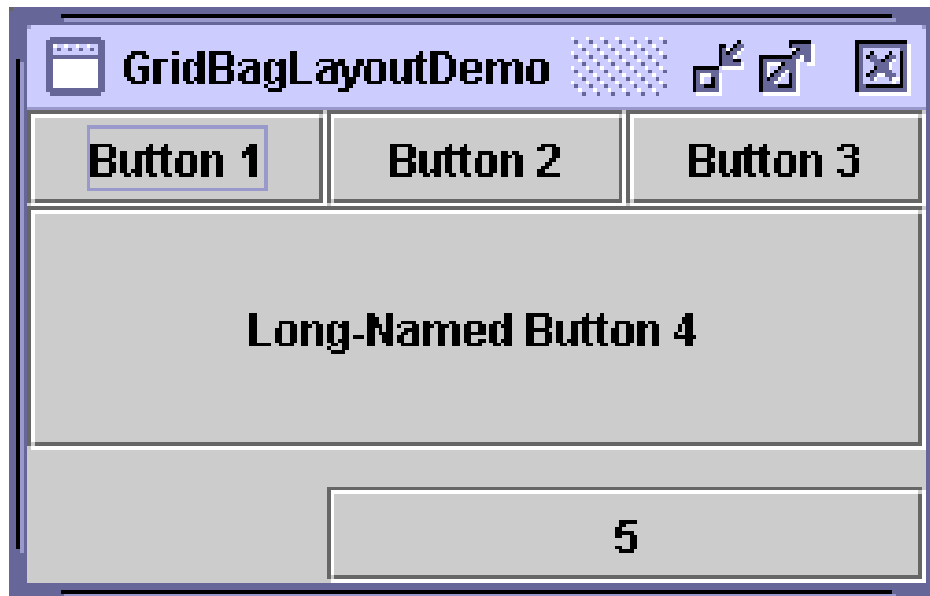
Klasės skirtos nustatyti GUI komponentų išdėstymą (4)

- GridLayout



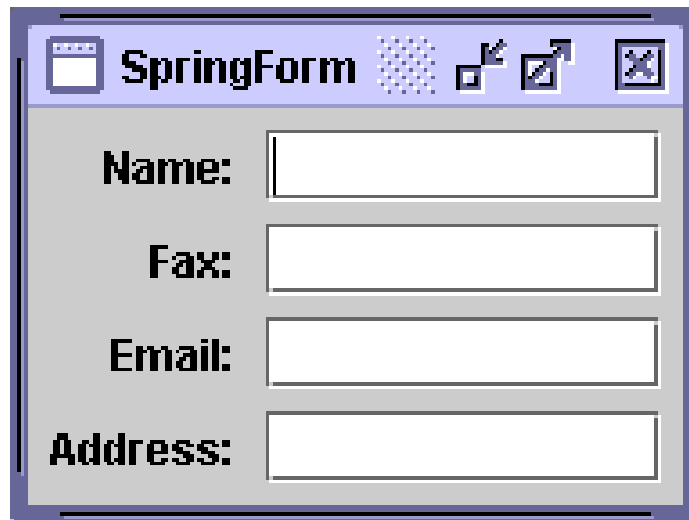
Klasės skirtos nustatyti GUI komponentų išdėstymą (5)

- GridBagLayout



Klasės skirtos nustatyti GUI komponentų išdėstymą (6)

- SpringLayout





Įvykių apdorojimas (1)

- Įvykis – tai veiksmas, kurį atlieka vartotojas, dirbdamas su grafine vartotojo sąsaja
- Kiekvieną įvykį aprašo jam skirtas objektas (pvz., `ActionEvent`, `MouseEvent`)
- Bet kuris Swing komponentas (šaltinis) gali fiksuoti su juo susijusius įvykius



Įvykių apdorojimas (2)

- Bet kuris Swing komponentas gali fiksuoti kelis skirtingo tipo įvykius
- Bet kuris “įvykio klausytojas” (event listener) gali būti užregistruotas keliems šaltiniams



Kai kurių įvykių pavyzdžiai

- `ActionEvent` – paspaudžiamas mygtukas, Enter klavišas ar pasirenkamas meniu punktas
- `WindowEvent` – vartotojas uždaro programos langą
- `MouseEvent` – vartotojas paspaudžia kurį nors pelės klavišą ant komponento
- `MouseEvent` – vartotojas užveda kursorių virš komponento

<http://java.sun.com/docs/books/tutorial/uiswing/events/eventsandcomponents.html>

<http://java.sun.com/docs/books/tutorial/uiswing/events/handling.html>



Įvykio apdorojimo realizavimas (1)

Įvykį apdorojanti klasė privalo:

a) arba pati implementuoti "įvykio klausytojo" interfeisą ir realizuoti visus jo metodus:

```
public class MyListener implements ActionListener {  
    public void actionPerformed(ActionEvent evt) {  
        // atitinkami veiksmai  
    }  
}
```



Įvykio apdorojimo realizavimas (2)

b) arba paveldėti “klausytojo” interfeisą realizuojančią klasę ir perdengti reikalingus metodus:

```
class MyMouseListener extends MouseAdapter {  
    public void mouseClicked(MouseEvent evt) {  
        // atitinkami veiksmai  
    }  
}
```



Įvykio apdorojimo realizavimas (3)

Naudojant *addXXXListener()* metodą, reikiams komponentams, užregistruojama įvykį apdorojanti klasė :

```
komponentas.addActionListener(new MyListener());  
    // variantui a)  
komponentas.addActionListener(this);  
    // variantui a), kai šaltinis ir apdorojimas vienoje  
    klasėje  
komponentas.addMouseListener(new MyMouseListener());  
    // variantui b)
```



Anoniminės “įvykio klausytojo” klasės panaudojimas

```
komponentas.addActionListener( new ActionListener() {  
    public void actionPerformed(ActionEvent evt) {  
        // atitinkami veiksmai  
    }  
});
```




Pastabos dėl įvykių apdorojimo

- Visi *xxxListener* interfeisai, turintys daugiau negu vieną metodą, turi specialias juos realizuojančias adapterių klases
- Įvykių klasės *xxxEvent* objektas turi metodus papildomai informacijai apie įvykį gauti
- Visos *xxxEvent* klasės paveldi *ObjectEvent* klasę, kurios metodas `getSource()` gražina įvykio šaltinį



Programos pavyzdys nr.2

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

public class Demo {
    public static void main(String args[]) {
        JFrame frame = new JFrame("Demo");
        frame.addWindowListener(
            new WindowAdapter() {
                public void windowClosing(WindowEvent evt) {
                    System.exit(0);
                }
            }
        );
    }
}

// tęsinys kitoje skaidrėje
```



Programos pavyzdys nr.2

```
JLabel label = new JLabel("Demonstracine programa");

JButton button = new JButton("Mygtukas");
button.setBackground(Color.RED);

button.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent evt) {
        JButton b = (JButton) evt.getSource();
        if (b.getBackground() == Color.RED) {
            b.setBackground(Color.GREEN);
        } else {
            b.setBackground(Color.RED);
        }
    }
});
```



Programos pavyzdys nr.2

```
        frame.getContentPane().add(label,  
                                   BorderLayout.CENTER);  
  
        frame.getContentPane().add(button,  
                                   BorderLayout.PAGE_END);  
  
        frame.setBounds(10, 10, 200, 100);  
        frame.setVisible(true);  
    }  
}
```



Paišymas

- Kiekvienas Swing'o komponentas turi metodą *paint(Graphics g)*, skirtą grafiniam to komponento atvaizdavimui
- Norint savaip atvaizduoti komponentą, reikia perdengti jo metodą *paint(Graphics g)* arba *paintComponent(Graphics g)*
- Komponento perpaišymas atliekamas metodo *repaint()* pagalba



Graphics ir Graphics2D objektai

- Tai objektai, turintys metodus, skirtus pašymui atlikti
- Saugo einamąjį grafinį kontekstą (pvz., pašymo spalva, šriftas, ploto dydis ir t.t.)

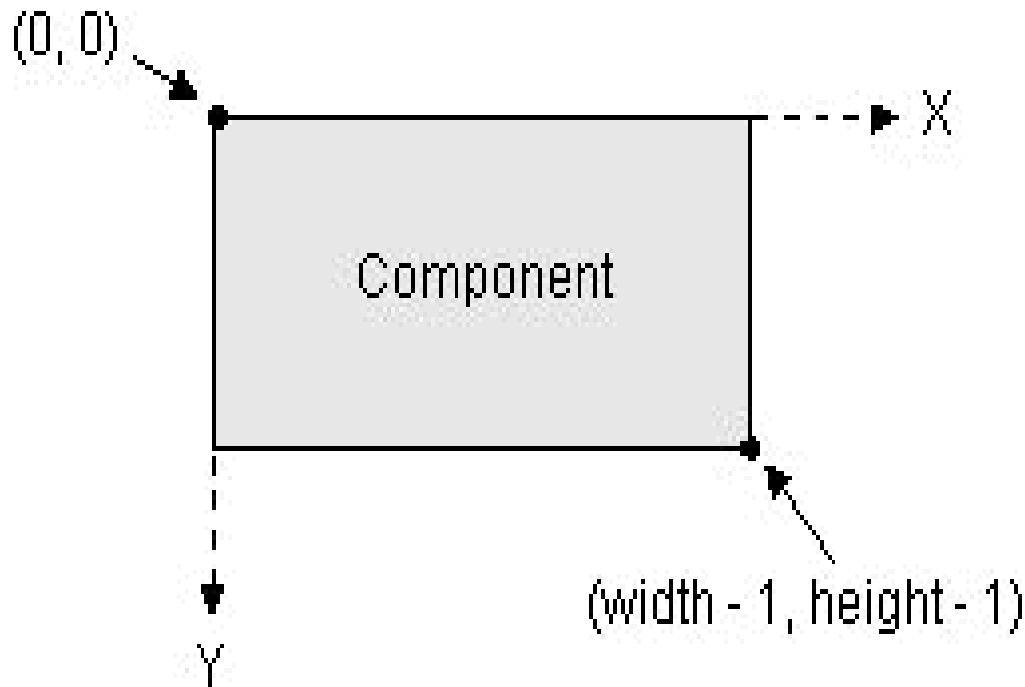


Graphics ir Graphics2D objektų galimybės

Šių objektų pagalba galima atvaizduoti:

- Įvairias geometrines figūras
- Tekstą
- Paveikslėlius

Swing komponentų koordinačių sistema



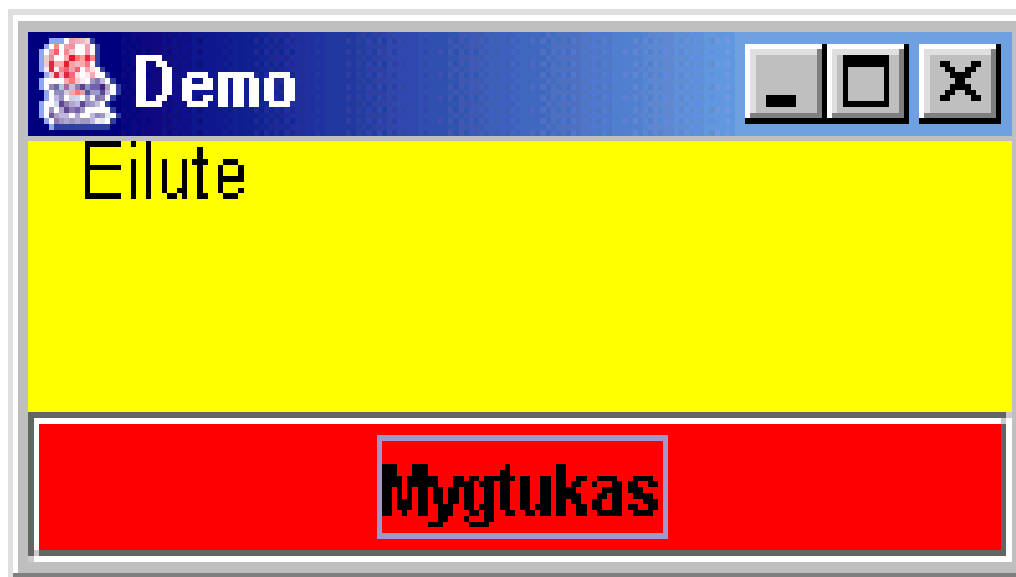


Programos pavyzdys nr.3

```
public class MyPanel extends JPanel {  
    public void paint(Graphics g) {  
        Graphics2D g2 = (Graphics2D) g;  
        g2.setBackground(Color.YELLOW);  
        g2.setFont(new Font("Arial", 0, 13));  
        g2.clearRect(0, 0,  
                     getWidth(),getHeight());  
        g2.setColor(Color.black);  
        g2.drawString("Eilute", 10, 10);  
    }  
}
```



Programos pavyzdys nr.3





Nuorodos (1)

- <http://java.sun.com/> - pagrindinis Java kūrėjos Sun Microsystems puslapis. Čia rasite visko...
- <http://java.sun.com/docs/books/tutorial/uiswing/index.html> - the Java Swing Tutorial.
- <http://java.sun.com/apis.html> - the Java API Documentation.
- <http://java.sun.com/docs/books/tutorial/uiswing/components/components.html> - Swing komponentų katalogas
- <http://java.sun.com/docs/books/tutorial/uiswing/events/index.html> - įvadas į įvykių apdorojimą.
- <http://java.sun.com/docs/books/tutorial/uiswing/events/handling.html> - populiariausių įvykių apdorojančių interfeisų realizacijos



Nuorodos (2)

- <http://java.sun.com/docs/books/tutorial/uiswing/layout/using.html> - komponentų išdėstymo būdai
- <http://www.javaworld.com/columns/jw-tips-index.shtml> - Java tips.
- <http://developer.java.sun.com/developer/JDCTechTips/> - Java Technical Tips