

ΓΡΑΦΙΚΕΣ ΔΙΕΠΑΦΕΣ ΧΡΗΣΤΗ- ΑWT

Γραφική διεπαφή χρήστη είναι ένα γραφικός τρόπος επικοινωνίας ενός προγράμματος με τον χρήστη του. Στηρίζεται στην έννοια των παραθύρων. Η επικοινωνία γίνεται μέσω μενού, κουμπιών κλπ.

Εργαλεία

- Χρήση εργαλειοθήκης `awt` (`abstract windowing tool`) (στάνταρντ: `java 1`) (σ' αυτό θ' αναφερθούμε εδώ)
- Χρήση εργαλειοθήκης `Swing` (εξέλιξη του `awt: java 2`)
- Είναι σύνολα κλάσεων για δημιουργία γραφικών διεπαφών χρήστη

ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΔΙΕΠΑΦΗΣ

- Συστατικά (components)
Δηλ. ετικέτες, πλήκτρα, μενού, λίστες κλπ που αποτελούν στιγμιότυπα κλάσεων.
- Υποδοχείς (containers)
Δηλ. γραφικές οντότητες που περιέχουν συστατικά. Π.χ. παράθυρα.
- Διαχειριστές διάταξης (layout managers)
Καθορίζουν τη διάταξη των συστατικών σ' ένα υποδοχέα.

ΔΗΜΙΟΥΡΓΙΑ ΔΙΕΠΑΦΗΣ

1. Δημιουργία υποδοχέα

(Δημιουργούμε το βασικό παράθυρο της εφαρμογής)

2. Δημιουργία συστατικών

(Δημιουργούμε τα συστατικά του παραθ.)

3. Καθορισμός διαχειριστή διάταξης

(Προσδιορίζουμε τον τρόπο διάταξης των συστατικών στο βασικό παράθυρο)

4. Προσθήκη συστατικών

(Εισάγουμε τα συστατικά στο παράθυρο)

ΓΡΑΦΙΚΕΣ ΔΙΕΠΑΦΕΣ ΧΡΗΣΤΗ- SWING (1)

Πλεονεκτήματα έναντι του awt

- Περισσότερα συστατικά
- Επεκτεταμένα χαρακτηριστικά συστατικών
- Καλύτερη εμφάνιση και αίσθηση
- Καλύτερος χειρισμός συμβάντων
- Πιό συμβατά προγράμματα στις διάφορες πλατφόρμες (συστατικά πλήρως υλοποιημένα σε java)

ΓΡΑΦΙΚΕΣ ΔΙΕΠΑΦΕΣ ΧΡΗΣΤΗ- SWING (2)

Ομοιότητες-Διαφορές από το awt

- Η βασική δομή του προγράμματος παραμένει .
- Οι βασικές έννοιες 'υποδοχέας', 'τομέας', 'συστατικό', και 'διαχειριστής διάταξης' παραμένουν .
- Η διαδικασία προσθήκης συστατικών διαφέρει .
- Η χρήση ενός συστατικού παραμένει ίδια .
- Αλλάζουν ελαφρώς τα ονόματα των κλάσεων (προσθήκη ενός «J» μπροστά συνήθως) .
- Πακέτο: `javax.swing`

ΔΗΜΙΟΥΡΓΙΑ ΥΠΟΔΟΧΕΑ (1)

Διαδικασία (ίδια)

- Δημιουργία υποκλάσης της JFrame.
- Δημιουργία στιγμιοτύπου της υποκλάσης στη μέθοδο main της υποκλάσης.

Δημιουργοί της JFrame

JFrame () → Δημιουργία παραθύρου χωρίς τίτλο

JFrame (String) → Δημιουργ. παραθ. με τίτλο

(Καλούνται μέσω του super απο τον δημιουργό της υποκλάσης)

ΔΗΜΙΟΥΡΓΙΑ ΥΠΟΔΟΧΕΑ (2)

Βασική δομή προγ/τος:

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

Συμπερίληψη εργαλειοθήκης

```
class MyApp extends JFrame {
```

Δημιουργός

```
    public MyApp (String title) {  
        super(title);
```

```
        <δημιουργία διαχειριστή διάταξης>
```

```
        <δημιουργία-προσθήκη συστατικών>
```

```
    }
```

Δημιουργία
στιγμιότυπου

```
public static void main() {
```

```
    MyApp app1 = new MyApp ("Application Window");
```

```
    <διαχείριση παραθύρου>} }
```

ΒΑΣΙΚΟΙ ΜΕΘΟΔΟΙ ΧΕΙΡΙΣΜΟΥ ΠΑΡΑΘΥΡΟΥ

pack() → μικρότερο δυνατό μέγεθος παραθ.

setVisible(boolean) → εμφάνιση/απόκρυψη
παραθύρου

setBounds(int, int, int, int) →
προσδιορισμός θέσης στην οθόνη
(x, y, w, h)

ΔΙΑΧΕΙΡΙΣΤΕΣ ΔΙΑΤΑΞΗΣ (1)

Διάταξη ροής (Κλάση: `FlowLayout`)

`FlowLayout()` → συστατικά το ένα μετά το άλλο (αριστ. → δεξιά) στο κέντρο

`FlowLayout(int)` → στοίχιση ανάλογα με το `int` (`FlowLayout.LEFT`/`FlowLayout.RIGHT`/`FlowLayout.CENTER`)

`FlowLayout(int, int, int)` → + οριζόντιο και κατακόρυφο διάκενο σε `pixels`
(2ο, 3ο `int`)

ΔΙΑΧΕΙΡΙΣΤΕΣ ΔΙΑΤΑΞΗΣ (2)

Διάταξη πλέγματος (Κλάση: GridLayout)

GridLayout(int, int) → συστατικά σε κελιά πλέγματος $\text{int} \times \text{int}$ (σε pixels). Τα συστατικά καταλαμβάνουν όλο τον χώρο των κελιών

GridLayout(int, int, int, int) → + οριζόντιο και κατακόρυφο διάκενο (3ο, 4ο int) σε pixels.

ΔΙΑΧΕΙΡΙΣΤΕΣ ΔΙΑΤΑΞΗΣ (3)

Περιφερειακή διάταξη (Κλάση: BorderLayout)

BorderLayout() → διάταξη χωρίς διάκενα

BorderLayout(int, int) →

οριζόντιο και κατακόρυφο διάκενο σε pixels



ΔΙΑΧΕΙΡΙΣΤΕΣ ΔΙΑΤΑΞΗΣ (4)

Καθορισμός Διαχειριστή Διάταξης

1. Δημιουργία διαχειριστή (δηλ. δημιουργία στιγμιοτύπου μέσω των δημιουργών των προηγούμενων κλάσεων)
2. Ενεργοποίηση διαχειριστή
Μέθοδος `setLayout (<layout instance>)`

ΔΙΑΧΕΙΡΙΣΤΕΣ ΔΙΑΤΑΞΗΣ

Καθορισμός Διαχειριστή Διάταξης

- Ισχύουν τα του awt
- Επιπλέον υπάρχει διαχειριστής εμφάνισης και αίσθησης

ΠΡΟΣΘΗΚΗ ΣΥΣΤΑΤΙΚΩΝ-ΤΟΜΕΑΣ ΠΕΡΙΕΧΟΜΕΝΟΥ (1)

- Η προσθήκη συστατικών δεν γίνεται απ' ευθείας στον κύριο υποδοχέα, αλλά σ' ένα ενδιάμεσο υποδοχέα, που ονομάζεται τομέας περιεχομένου (content pane)
- Βασικά, ο κύριος υποδοχέας (πλαίσιο παραθύρου) αποτελείται από διάφορους τομείς. Ο κεντρικός τομέας είναι ο τομέας περιεχομένου

ΠΡΟΣΘΗΚΗ ΣΥΣΤΑΤΙΚΩΝ-ΤΟΜΕΑΣ ΠΕΡΙΕΧΟΜΕΝΟΥ (2)

Διαδικασία

1. Δημιουργία ενός (στιγμιοτύπου) τομέα (δημιουργός κλάσης `JPanel`).
2. Δημιουργία και προσθήκη συστατικών στον τομέα αυτό (`add(Component)`).
3. Προσθήκη του τομέα (χρήση `add`) ή καθορισμός του τομέα ως τομέα περιεχομένου (`setContentPane(Container)`)

Όλα τα συστατικά έχουν κοινή υπερκλάση την `JComponent`, απ' όπου κληρονομούν μεθόδους (`setEnabled`, `setVisible`, `setSize` κλπ)

ΚΑΘΟΡΙΣΜΟΣ ΑΙΣΘΗΣΗΣ ΚΑΙ ΕΜΦΑΝΙΣΗΣ (1)

- Γίνεται μέσω της κλάσης UIManager
- Επιλογή από τρεις τύπους
 - Τύπος Windows
 - Τύπος Motif X-Windows
 - Τύπος Metal (java)
- Η επιλογή γίνεται από τη μέθοδο `setLookAndFeel(LookAndFeel)`
- Δημιουργία στιγμιοτύπου LookAndFeel:
 - (α) `getCrossPlatformLookAndFeelClassName()`
(για επιλογή metal)
 - (β) `getSystemLookAndFeelClassName()`
(για επιλογή αίσθησης & εμφάνισης του λειτουργικού συστήματος που χρησιμοποιείται)

ΚΑΘΟΡΙΣΜΟΣ ΑΙΣΘΗΣΗΣ ΚΑΙ ΕΜΦΑΝΙΣΗΣ (2)

Η `setLookAndFeel` χρειάζεται χειρισμό εξαιρέσεων.

```
try {  
    UIManager.setLookAndFeel (UIManager.getCrossPlatform  
    LookAndFeelClassName () ; }  
catch (Exception e) {  
    System.err.println("Can't set look and feel: "+e);  
}
```

ΕΤΙΚΕΤΕΣ

Κλάση: JLabel

Δημιουργοί:

```
JLabel (String)
```

```
JLabel (String, int)
```

(όπου int → SwingConstants.LEFT/CENTER/RIGHT)

```
JLabel (String, Icon, int)
```

Δημιουργία εικονιδίου

```
ImageIcon ic1 = new ImageIcon("icon1.gif");
```

ΠΛΗΚΤΡΑ/ΚΟΥΜΠΙΑ

Κλάση: JButton

Δημιουργοί:

JButton (String)

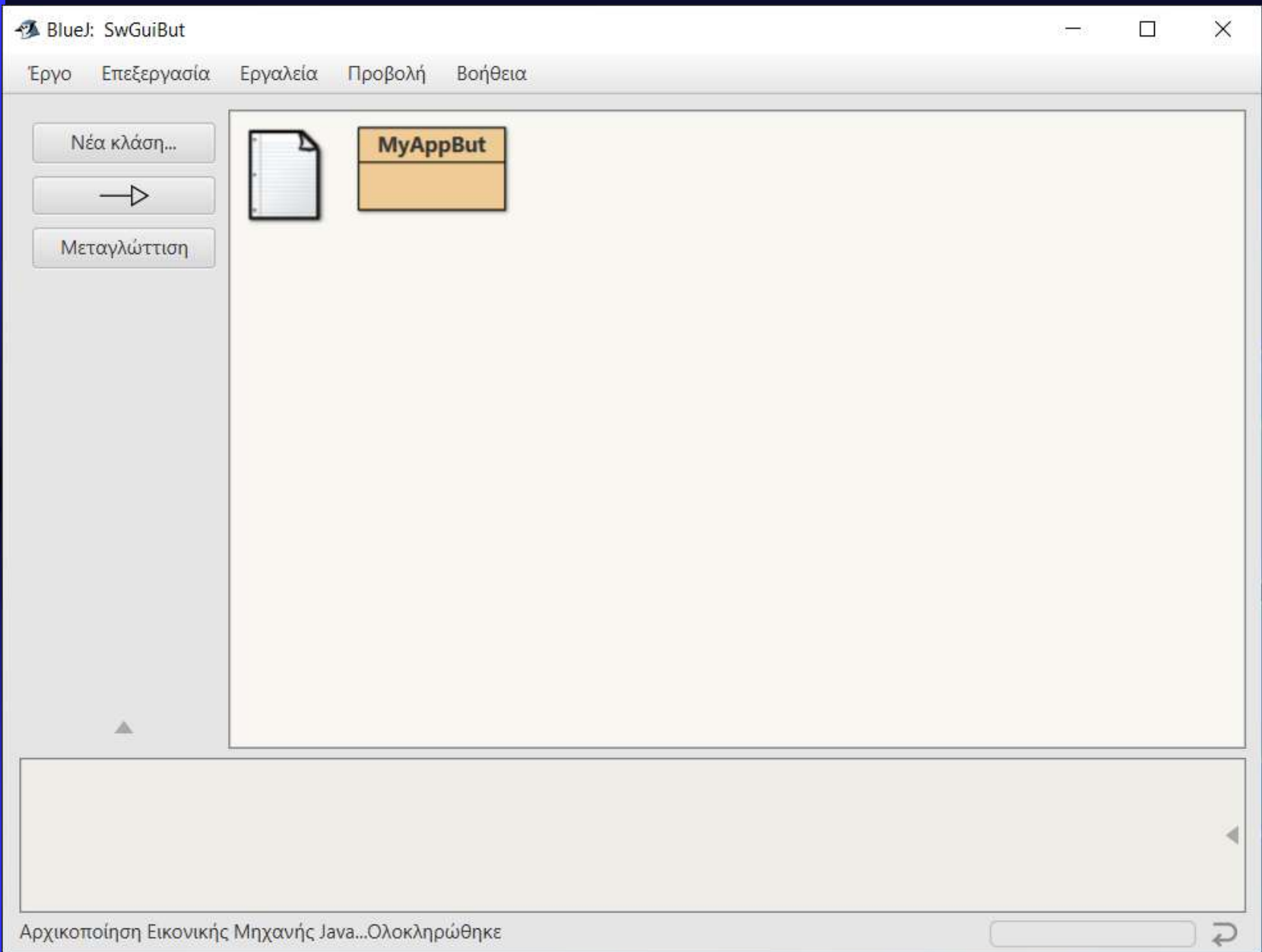
JButton (Icon)

JButton (String, Icon)

Μέθοδοι:

setLabel (String)

getLabel ()



MyAppBut - SwGuiBut

Κλάση Επεξεργασία Εργαλεία Επιλογές

MyAppBut X

Μεταγλώττιση Αναίρεση Αποκοπή Αντιγραφή Επικόλληση Εύρεση... Κλείσιμο Υλοποίηση

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

public class MyAppBut extends JFrame {
    public MyAppBut (String title) {
        super(title);
        Font f1 = new Font("Helvetica", Font.BOLD, 14);
        setFont(f1);

        FlowLayout lay = new FlowLayout(FlowLayout.CENTER, 10, 10);
        String butLab1 = "This is a button-Press it!!!";
        String butLab2 = "This is another button-Don't Press it!!!";
        JButton firstButton = new JButton(butLab1);
        JButton secondButton = new JButton(butLab2);

        JPanel frame = new JPanel();
        frame.setBackground(Color.red);
        frame.setLayout(lay);
        frame.add(firstButton);
        frame.add(secondButton);
        // setContentPane(frame);
        add(frame);
    }
}
```

Η κλάση μεταγλωττίστηκε - δεν υπήρχαν συντακτικά λάθη

αποθηκευμένο

MyAppBut - SwGuiBut

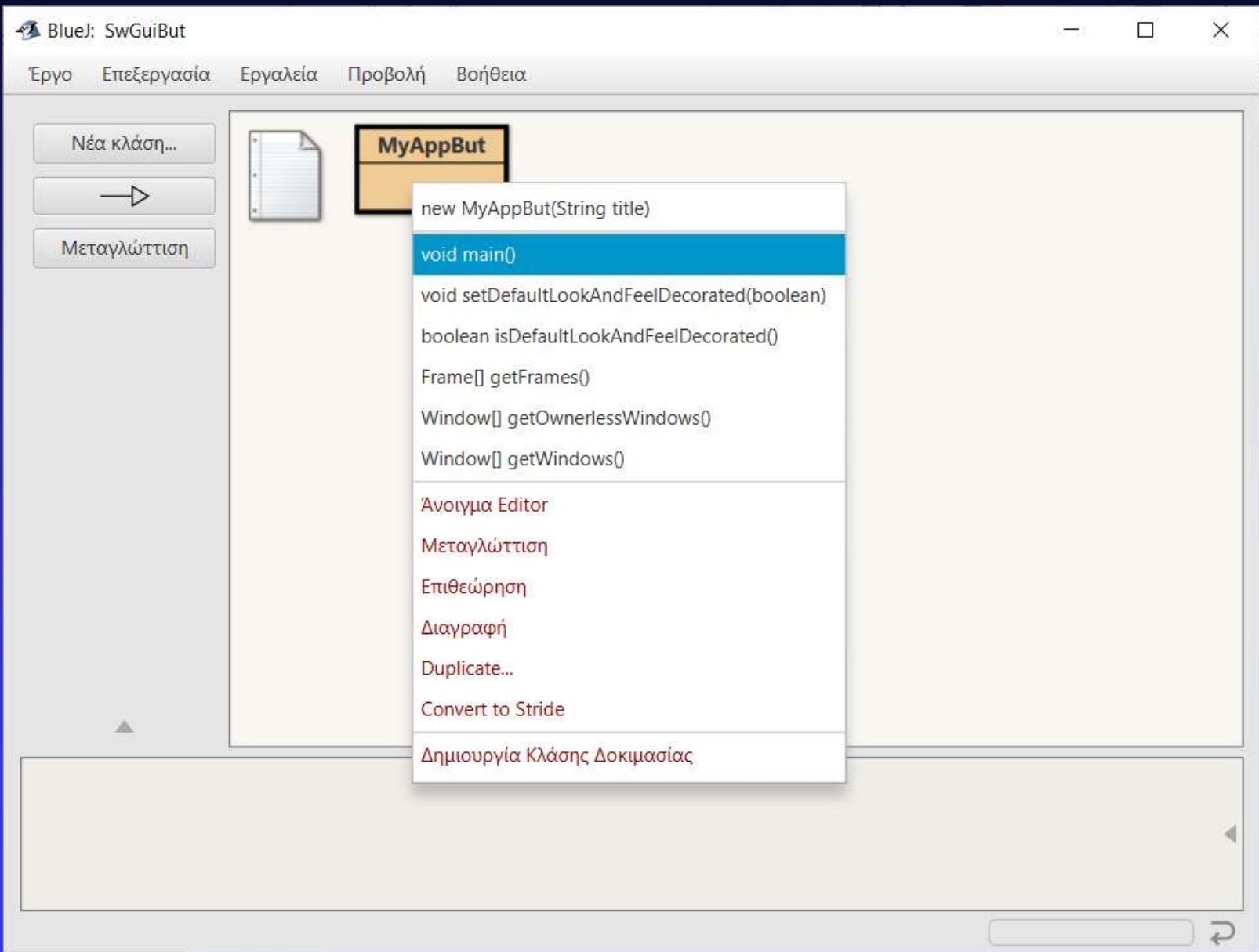
Κλάση Επεξεργασία Εργαλεία Επιλογές

MyAppBut X

Μεταγλώττιση Αναίρεση Αποκοπή Αντιγραφή Επικόλληση Εύρεση... Κλείσιμο Υλοποίηση

```
public static void main(){  
    try {  
        UIManager.setLookAndFeel(UIManager.getCrossPlatformLookAndFeelClassName());  
        // UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());  
    }  
    catch (Exception e) {  
        System.err.println("Can't set look and feel: " + e);  
    }  
  
    MyAppBut app = new MyAppBut("Application Window");  
  
    WindowListener I = new WindowAdapter() {  
        public void windowClosing(WindowEvent e) {  
            System.exit(0);  
        }  
    };  
    app.addWindowListener(I);  
    // app.pack();  
    // app.setBounds(50, 40, 460, 130);  
    app.setSize(500, 200);  
    app.setVisible(true);  
}
```

αποθηκευμένο





Application Window



This is a button-Press it!!!

This is another button-Don't Press it!!!

ΠΛΑΙΣΙΑ ΕΛΕΓΧΟΥ

Κλάση: JCheckbox

Δημιουργοί:

JCheckbox (String)

JCheckbox (String, boolean)

JCheckbox (Icon)

JCheckbox (Icon, boolean)

JCheckbox (String, Icon)

JCheckbox (String, Icon, boolean)

Μέθοδοι:

setEnabled (boolean)

ΠΑΡΑΔΕΙΓΜΑ

Βήμα 1

```
ButtonGroup lang = new ButtonGroup();
```

Βήμα 2

```
JCheckbox c1 = new JCheckbox ("Pascal", false);
```

```
JCheckbox c2 = new JCheckbox ("Java", false);
```

Βήμα 3

```
lang.add(c1);
```

```
lang.add(c2);
```

ΚΟΥΜΠΙΑ ΕΠΙΛΟΓΗΣ

Κλάση: JRadioButton

Δημιουργοί

JRadioButton (String)

JRadioButton (String, boolean)

JRadioButton (Icon)

JRadioButton (Icon, boolean)

JRadioButton (String, Icon)

JRadioButton (String, Icon, boolean)

Μέθοδοι:

setEnabled (boolean)

ΟΜΑΔΟΠΟΙΗΜΕΝΑ ΠΛΑΙΣΙΑ ΕΛΕΓΧΟΥ/ΚΟΥΜΠΙΑ ΕΠΙΛΟΓΗΣ

Κλάση: ButtonGroup

Δημιουργός: ButtonGroup ()

Διαδικασία:

1. Δημιουργία στιγμιοτύπου ButtonGroup
2. Δημιουργία πλαισίων/κουμπιών
3. Προσθήκη πλαισίων/κουμπιών:
add (Component)

ΛΙΣΤΕΣ ΕΠΙΛΟΓΗΣ (1)

Κλάση: JComboBox

Δημιουργός: JComboBox()

Διαδικασία:

1. Δημιουργία σύνθετου πλαισίου

```
JComboBox langBox = new JComboBox();
```

2. Προσθήκη στοιχείων

```
langBox.addItem("Pascal");
```

3. Μετατροπή σύνθετου πλαισίου σε λίστα

```
setEditable(false);
```

ΛΙΣΤΕΣ ΕΠΙΛΟΓΗΣ (2)

Μέθοδοι :

`getItem(int)`

Επιστρέφει το στοιχείο στη θέση `int` (πρώτη θέση: 0)

`getItemCount()`

Επιστρέφει τον αριθμό των στοιχείων της λίστας

`setSelectedIndex(int)`

Επιλέγει το στοιχείο στη θέση `int`

`getSelectedIndex()`

Επιστρέφει τη θέση του τρέχοντος επιλεγμένου στοιχείου

`getSelectedItem()`

Επιστρέφει το όνομα του τρέχοντος επιλεγμένου στοιχείου

ΓΡΑΜΜΕΣ ΚΥΛΙΣΗΣ

Κλάση: JScrollBar

Επιλογή τιμής με ολίσθηση ενός πλαισίου.

Δημιουργοί:

JScrollBar(int)

(int → SwingConstants.HORIZONTAL/VERTICAL)

JScrollBar(int, int, int, int, int)

(int → SwingConstants.HORIZONTAL/VERTICAL)

(int → αρχική τιμή)

(int → μέγεθος πλαισίου κύλισης)

(int → ελάχιστη τιμή)

(int → μέγιστη τιμή)

ΠΕΔΙΑ ΚΕΙΜΕΝΟΥ (1)

Κλάση: JTextField, JPasswordField

Δημιουργοί:

JTextField(int)

JTextField(String, int)

JPasswordField(int)

JPasswordField(String, int)

(Χρήση της setEchoChar(char))

ΠΕΡΙΟΧΕΣ ΚΕΙΜΕΝΟΥ

Κλάση: JTextArea

Δημιουργοί:

JTextArea(int, int)

JTextArea(String)

JTextArea(String, int, int)

ToolBar - SwingExamples1

Κλάση Επεξεργασία Εργαλεία Επιλογές

ToolBar x

Μεταγλώττιση Αναίρεση Αποκοπή Αντιγραφή Επικόλληση Εύρεση... Κλείσιμο Υλοποίηση

```
import java.awt.*;
import javax.swing.*;
public class ToolBar extends JFrame {
    public ToolBar() {
        super("ToolBar");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        ImageIcon image1 = new ImageIcon("button1.gif");
        JButton button1 = new JButton("dsfsd", image1);
        ImageIcon image2 = new ImageIcon("button2.gif");
        JButton button2 = new JButton(image2);
        ImageIcon image3 = new ImageIcon("button3.gif");
        JButton button3 = new JButton(image3);
        JToolBar bar = new JToolBar();
        bar.add(button1);
        bar.add(button2);
        bar.add(button3);
        JTextArea edit = new JTextArea(8,40);
        JScrollPane scroll = new JScrollPane(edit);
        JPanel pane = new JPanel();
        BorderLayout bord = new BorderLayout();
        pane.setLayout(bord);
        pane.add("North", bar);
        pane.add("Center", scroll);

        setContentPane(pane);
    }
    public static void main() {
        ToolBar frame = new ToolBar();
        frame.pack();
        frame.setVisible(true);
    }
}
```

αποθηκευμένο



ΝΕΑ ΣΤΟΙΧΕΙΑ: ΠΡΟΤΥΠΑ ΠΑΡΑΘΥΡΑ ΔΙΑΛΟΓΟΥ

Κλάση: JOptionPane

- Υπάρχουν τέσσερις τύποι παραθύρων διαλόγου
 - Επιβεβαίωσης (ConfirmDialog)
 - Εισόδου (InputDialog)
 - Μηνύματος (MessageDialog)
 - Επιλογής (OptionDialog)

ΝΕΑ ΣΤΟΙΧΕΙΑ:

ΠΑΡΑΘΥΡΑ ΕΠΙΒΕΒΑΙΩΣΗΣ (1)

- Ερώτηση με κουμπιά Yes, No, Cancel

Μέθοδος1:

`showConfirmDialog(Component, Object)`

Component: ο υποδοχέας (null: κέντρο οθόνης)

Object: String, συστατικό ή Icon

Επιστρέφει: μία από τρεις ακέραιες τιμές
(YES_OPTION, NO_OPTION, CANCEL_OPTION)

Π.χ.

```
int answer = JOptionPane.showConfirmDialog(null,  
    "Are you sure to delete the file?");
```

ΠΑΡΑΘΥΡΑ ΕΠΙΒΕΒΑΙΩΣΗΣ (2)

Μέθοδος2:

```
showConfirmDialog(Component, Object,  
                  String, int, int)
```

String: κείμενο γραμμής τίτλου παραθύρου

int: YES_NO_CANCEL_OPTION, YES_NO_OPTION

int: ERROR_MESSAGE, INFORMATION_MESSAGE,
PLAIN_MESSAGE, QUESTION_MESSAGE,
WARNING_MESSAGE

Π.χ.

```
int answer = JOptionPane.showConfirmDialog(null,  
    "Error reading file", "File Input Error",  
    JOptionPane.YES_NO_OPTION, JOptionPane.  
    ERROR_MESSAGE);
```

ΝΕΑ ΣΤΟΙΧΕΙΑ:

ΠΑΡΑΘΥΡΑ ΕΙΣΟΔΟΥ

- Ερώτηση με πεδίο κειμένου για απάντηση

Μέθοδος1:

```
showInputDialog(Component, Object)
```

Επιστρέφει: String (το κείμενο της απάντησης)

Π.χ.

```
string answer = JOptionPane.showInputDialog(null,  
    "Enter your title:");
```

Μέθοδος2:

```
showInputDialog(Component, Object, String,  
    int)
```

Π.χ.

```
string answer = JOptionPane.showInputDialog(null,  
    "Enter your title:", "Enter Title", JOptionPane.  
    QUESTION_MESSAGE);
```

ΝΕΑ ΣΤΟΙΧΕΙΑ:

ΠΑΡΑΘΥΡΑ ΜΗΝΥΜΑΤΟΣ

- Εμφάνιση πληροφορίας

Μέθοδος1:

```
showMessageDialog(Component, Object)
```

Επιστρέφει: Δεν επιστρέφει τιμή

Π.χ.

```
JOptionPane.showMessageDialog(null, "Title is  
missing");
```

Μέθοδος2:

```
showMessageDialog(Component, Object,  
String, int)
```

Π.χ.

```
int answer = JOptionPane.showInputDialog(null, "The  
title is missing", "Missing Component Message",  
JOptionPane.WARNING_MESSAGE);
```


ΝΕΑ ΣΤΟΙΧΕΙΑ:

ΠΑΡΑΘΥΡΑ ΕΠΙΛΟΓΗΣ (1)

- Πιο περίπλοκο από τα άλλα
- Συνδυάζει χαρακτηριστικά όλων των άλλων

Μέθοδος:

```
showOptionDialog(Component, Object,  
String, int, int, Icon, Object[], Object)
```

Icon: στιγμιότυπο Icon αντί του υπάρχοντος

Object[]: πίνακας που περιέχει τα συστατικά που είναι οι επιλογές του παραθύρου.

Object: Η προεπιλεγμένη επιλογή, αν δεν χρησιμοποιηθούν τα YES_NO_CANCEL_OPTION, YES_NO_OPTION

Επιστρέφει: int

ΠΑΡΑΘΥΡΑ ΕΠΙΛΟΓΗΣ (2)

Π.χ.

```
Jbutton gender [] = new Jbutton[3];  
gender[0] = new Jbutton("Male");  
gender[1] = new Jbutton("Female");  
gender[2] = new Jbutton("None");  
  
int answer = JOptionPane.showInputDialog(null,  
"What is your gender?", "Gender", 0,  
JOptionPane.INFORMATION_MESSAGE, null, gender,  
gender[2]);
```

ΧΕΙΡΙΣΜΟΣ ΣΥΜΒΑΝΤΩΝ

- Μια κλάση που θέλει να αποκρίνεται σε συμβάντα πρέπει να υλοποιεί αντίστοιχη διεπαφή (interface), που λέγεται ακροατής συμβάντων (event listener).
- Κάθε ακροατής χειρίζεται ένα συγκεκριμένο είδος συμβάντος
- Μια κλάση μπορεί να υλοποιήσει όσους ακροατές χρειάζεται.

Διαδικασία

1. Υλοποίηση ακροατή συμβάντων
2. Συσχέτιση όποιων συστατικών επιθυμούμε με ένα ή περισσότερους ακροατές συμβάντων

ΒΑΣΙΚΟ ΠΛΑΙΣΙΟ ΠΡΟΓ/ΤΟΣ

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

```
import java.awt.event;
```

Πακέτο χειρισμού συμβάντων.

```
class MyApp extends JFrame {  
    public MyApp (String title) {  
        super(title);  
        <δημιουργία διαχειριστή διάταξης>  
        <δημιουργία-προσθήκη συστατικών>}  
}
```

Κώδικας χειρισμού
προτύπων συμβάντων:
κλείνει την εφαρμογή
όταν κλείνει το
πλαίσιο (παράθυρο).

```
Public static void main() {  
    MyApp app = new MyApp (“Application Window”);  
    WindowListener L = new WindowAdapter() {  
        public void windowClosing(WindowEvent e) {  
            System.exit(0);}}};  
    app.addWindowListener(L);  
    app.pack();  
    app.setVisible(true);} }
```

ΑΚΡΟΑΤΕΣ ΣΥΜΒΑΝΤΩΝ (1)

- **ActionListener** (Συμβάντα ενέργειας: παράγονται από ενέργεια σε συστατικό, π.χ. πάτημα σε κουμπί)
- **AdjustmentListener** (Συμβάντα ρύθμισης: παράγονται από ρύθμιση συστατικού π.χ. μετακίνηση γραμμής κύλισης)
- **FocusListener** (Συμβάντα εστίασης: παράγονται από συστατικό που παίρνει ή χάνει την εστίαση, π.χ. πεδίο κειμένου)
- **ItemListener** (Συμβάντα στοιχείου: παράγονται όταν ένα στοιχείο, π.χ. σε πλαίσιο ελέγχου, αλλάζει)
- **KeyListener** (Συμβάντα πληκτρολογίου: παράγονται όταν εισάγουμε δεδομένα από το πληκτρολόγιο)

ΑΚΡΟΑΤΕΣ ΣΥΜΒΑΝΤΩΝ (2)

- **MouseListener** (Συμβάντα ποντικιού: παράγονται από πατήματα του ποντικιού)
- **MouseMotionListener** (Συμβάντα κίνησης ποντικιού: παράγονται από κίνηση του ποντικιού σε συστατικό)
- **TextListener** (Συμβάντα κειμένου: παράγονται από μεταβολές σε κείμενο)
- **WindowListener** (Συμβάντα παραθύρου: παράγονται από μεταβολές σε παράθυρο, π.χ. ελαχιστοποίηση)

ΣΥΣΧΕΤΙΣΗ ΑΚΡΟΑΤΩΝ

ΣΥΜΒΑΝΤΩΝ ΜΕ ΣΥΣΤΑΤΙΚΑ (1)

- `addActionListener()` (Jbutton, JCheckBox, JTextField, JRadioButton)
- `addAdjustmentListener()` (JScrollBar)
- `addFocusListener()` (όλα του Swing)
- `addItemListener()` (Jbutton, JCheckBox, JRadioButton)
- `addKeyListener` (όλα του Swing)
- `addMouseListener` (όλα του Swing)
- `addMouseMotionListener` (όλα του Swing)
- `addTextListener` (JTextField)
- `addWindowListwner` (όλα των Jwindow, JFrame)

ΣΥΣΧΕΤΙΣΗ ΑΚΡΟΑΤΩΝ

ΣΥΜΒΑΝΤΩΝ ΜΕ ΣΥΣΤΑΤΙΚΑ (2)

Όλες οι μέθοδοι `add..` παίρνουν ένα όρισμα: το αντικείμενο που «ακούει» τα συμβάντα αυτού του είδους. Η χρήση του `this`, στην περίπτωση αυτή, δηλώνει την τρέχουσα κλάση ως ακροατή.

Π.χ.

```
Jbutton b = new Jbutton("button");  
b.addActionListener(this);
```

Μπορούμε να δηλώσουμε και ένα διαφορετικό αντικείμενο, αρκεί η κλάση του να υλοποιεί τη σωστή διεπαφή ακροατή.

ΜΕΘΟΔΟΙ ΧΕΙΡΙΣΜΟΥ ΣΥΜΒΑΝΤΩΝ

- Όταν συσχετίζουμε ένα ακροατή συμβάντων (διεπαφή) με μια κλάση, τότε η κλάση αυτή πρέπει να υλοποιεί όλες τις μεθόδους της αντίστοιχης διεπαφής.
- Κάθε τέτοια μέθοδος καλείται αυτόματα από το παραθυρικό σύστημα όταν συμβεί αντίστοιχο συμβάν.

ActionListener

ActionListener

Έχει μια και μοναδική μέθοδο, την `actionPerformed()`. Κάθε κλάση που υλοποιεί την `ActionListener` πρέπει να υλοποιεί την παρακάτω μέθοδο:

```
public void actionPerformed(ActionEvent e){  
    <χειρισμός συμβάντος>  
}
```

Η `ActionEvent` είναι υποκλάση της `EventObject` (πακέτο `java.awt.event`)

getSource ()

Μπορεί να χρησιμοποιηθεί για να προσδιορίσουμε το συστατικό στο οποίο στάλθηκε ένα συμβάν (ή με άλλα λόγια το συστατικό που «άκουσε» το συμβάν).

Π.χ.

```
public void actionPerformed(ActionEvent, e){  
    Object comp = e.getSource();  
    if (comp == quitButton)  
        quitProgram();  
    else if (comp == sortRecords)  
        sortRecords();  
}
```

instanceof

Μπορεί να χρησιμοποιηθεί για να έλεγχο του είδους του συστατικού:

Π.χ.

```
public void actionPerformed(ActionEvent, e){  
    Object comp = e.getSource();  
    if (comp instanceof JTextField)  
        calculateScore();  
    else if (comp instanceof JButton)  
        quitProgram();  
}
```

AdjustmentListener

Έχει μια και μοναδική μέθοδο, την

```
public void adjustmentValueChanged(AdjustmentEvent e){  
<χειρισμόςσυμβάντος>  
}
```

FocusListener

Μέθοδοι διασύνδεσης :

```
public void FocusGained(FocusEvent e) {  
<χειρισμός συμβάντος>  
}
```

```
public void FocusLost(FocusEvent e) {  
<χειρισμός συμβάντος>  
}
```

ItemListener

JButton, JCheckBox, JComboBox, JRadioButton

Όταν ένα στοιχείο επιλέγεται ή αποεπιλέγεται

Μέθοδος διασύνδεσης:

```
public void itemStateChanged(ItemEvent e) {  
<χειρισμόςσυμβάντος>  
}
```

Για να καθορίσετε το στοιχείο όπου συνέβη το συμβάν: μέθοδος `getItem()` στο αντικείμενο `ItemEvent`.

KeyListener

Μέθοδοι διασύνδεσης:

```
public void keyPressed(KeyEvent e) {  
<χειρισμόςσυμβάντος>  
}  
public void keyReleased(KeyEvent e) {  
<χειρισμόςσυμβάντος>  
}  
public void keyTyped(KeyEvent e) {  
<χειρισμόςσυμβάντος>  
}
```

Χρήση μεθόδου `getKeyChar()` του `KeyEvent`

MouseListener

Μέθοδοι διασύνδεσης:

```
public void mouseClicked(MouseEvent e)
public void mouseEntered(MouseEvent e)
public void mouseExited(MouseEvent e)
public void mousePressed(MouseEvent e)
public void mouseReleased(MouseEvent e)
```

Χρήση μεθόδου `getPoint()` του `MouseEvent`

MouseEventListener

Μέθοδοι διασύνδεσης:

```
public void mouseDragged(MouseEvent e)  
public void mouseMoved(MouseEvent e)
```

Χρήση μεθόδων του MouseEvent

WindowListener

Μέθοδοι διασύνδεσης:

```
public void windowActivated(WindowEvent e)
public void windowClosed(WindowEvent e)
public void windowClosing(WindowEvent e)
public void windowDeactivated(WindowEvent e)
public void windowDeiconified(WindowEvent e)
public void windowIconified(WindowEvent e)
public void windowOpened(WindowEvent e)
```

Slider - SwingExamples1

Κλάση Επεξεργασία Εργαλεία Επιλογές

Slider X

Μεταγλώττιση Αναίρεση Αποκοπή Αντιγραφή Επικόλληση Εύρεση... Κλείσιμο Υλοποίηση

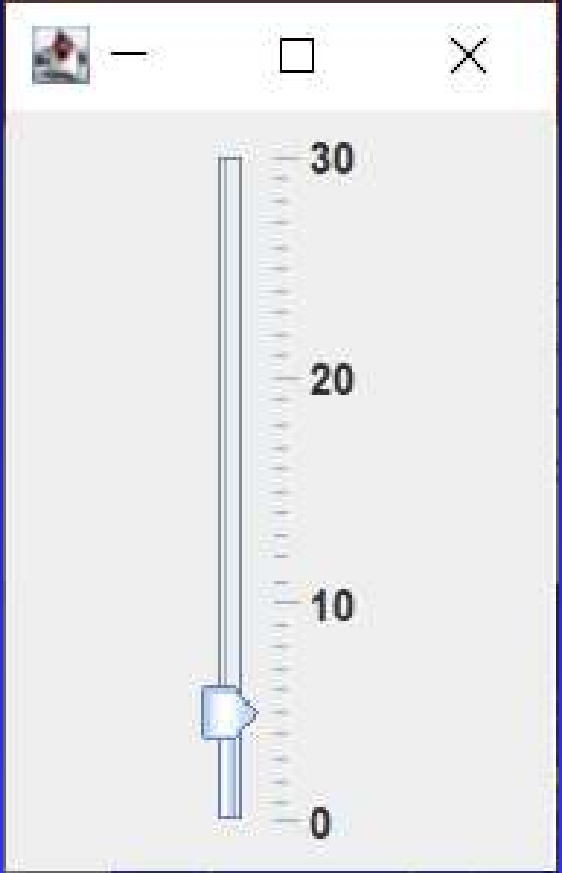
```
import java.awt.event.*;
import javax.swing.*;

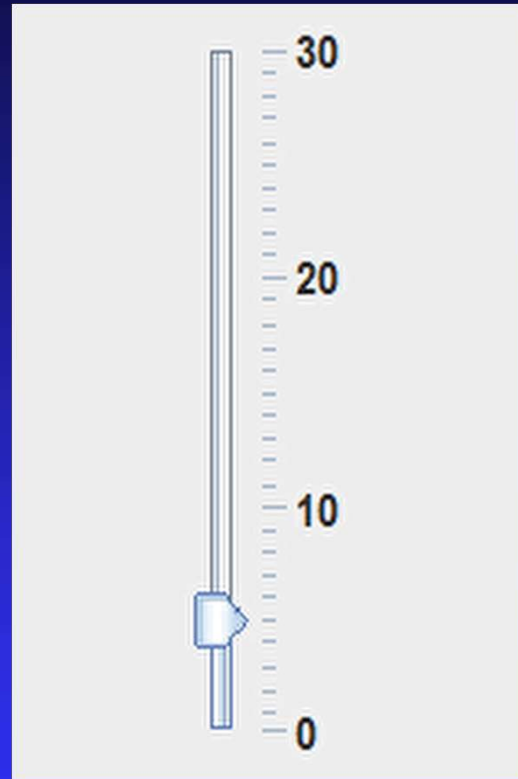
public class Slider extends JFrame {
    public Slider() {
        super("Slider");
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JSlider pickNum = new JSlider(JSlider.VERTICAL, 0, 30, 5);
        pickNum.setMajorTickSpacing(10);
        pickNum.setMinorTickSpacing(1);
        pickNum.setPaintTicks(true);
        pickNum.setPaintLabels(true);
        JPanel pane = new JPanel();
        pane.add(pickNum);

        setContentPane(pane);
    }

    public static void main() {
        // UIManager.setLookAndFeel(com.sun.java.swing.plaf.motif.MotifLookAndFeel);
        Slider frame = new Slider();
        frame.pack();
        frame.setVisible(true);
    }
}
```

αποθηκευμένο





MyAppDial1 - SwGuiDial

Κλάση Επεξεργασία Εργαλεία Επιλογές

MyAppDial1 X

Μεταγλώττιση Αναίρεση Αποκοπή Αντιγραφή Επικόλληση Εύρεση... Κλείσιμο Υλοποίηση

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class MyAppDial1 extends JFrame {
    public MyAppDial1 (String title) {
        super(title);

        JPanel frame = new JPanel();

        String answer = JOptionPane.showInputDialog(null, "Enter your title", "Enter Title",
            JOptionPane.QUESTION_MESSAGE);
        setContentPane(frame);
    }

    public static void main(){
        try {
            UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
        } catch (Exception e) {
            System.err.println("Can't set loo and feel: " + e);
        }
        MyAppDial1 app = new MyAppDial1("Application Window");

        WindowListener I = new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        };
        app.addWindowListener(I);
        app.pack();
        app.setVisible(true);
    }
}
```

αποθηκευμένο

Enter Title



Enter your title

OK

Cancel

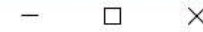


Enter your title

OK

Cancel

MyAppEv1 - SwGuiEv1



Κλάση Επεξεργασία Εργαλεία Επιλογές

MyAppEv1 X

Μεταγλώττιση Αναίρεση Αποκοπή Αντιγραφή Επικόλληση Εύρεση... Κλείσιμο Υλοποίηση

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

public class MyAppEv1 extends JFrame implements ActionListener{
    JButton b1 = new JButton("PA0");
    JButton b2 = new JButton("OSFP");
    public MyAppEv1 (String title) {
        super(title);
        b1.addActionListener(this);
        b2.addActionListener(this);
        JPanel frame = new JPanel();
        frame.add(b1);
        frame.add(b2);
        setContentPane(frame);
    }

    public static void main(){
        try {
            UIManager.setLookAndFeel(UIManager.getCrossPlatformLookAndFeelClassName());
        } catch (Exception e) {
            System.err.println("Can't set loo and feel: " + e);
        }
        MyAppEv1 app = new MyAppEv1("Application Window");

        WindowListener L = new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        };
        app.addWindowListener(L);
        app.pack();
        app.setVisible(true);
    }
}
```

αποθηκευμένο





