

Αντικειμενοστρεφής Προγραμματισμός

Διάλεξη – 12 : **ΒΑΣΙΚΑ ΣΤΟΙΧΕΙΑ GUI AWT ΚΑΙ SWING**

Βασικά στοιχεία ενός GUI

- Ένα Graphical User Interface-GUI (στα ελληνικά Γραφική Διεπαφή με το Χρήστη) είναι το μέρος του προγράμματος, που φροντίζει για τον τρόπο εμφάνισης και χειρισμού του προγράμματος από τον χρήστη.
- Ένα GUI αποτελείται από GUI-components. Οι κλάσεις που χρησιμοποιούνται για την κατασκευή αντικειμένων τύπου GUI-components ανήκουν στο java.awt (Abstract Windowing Toolkit) package.
- Οι βασικότερες από αυτές είναι η κλάση Component και η κλάση Container.
- Κάθε κλάση που κληρονομεί την κλάση Component είναι και αυτή ένα Component. Επίσης κάθε κλάση που κληρονομεί την κλάση Container είναι και αυτή ένα Container.

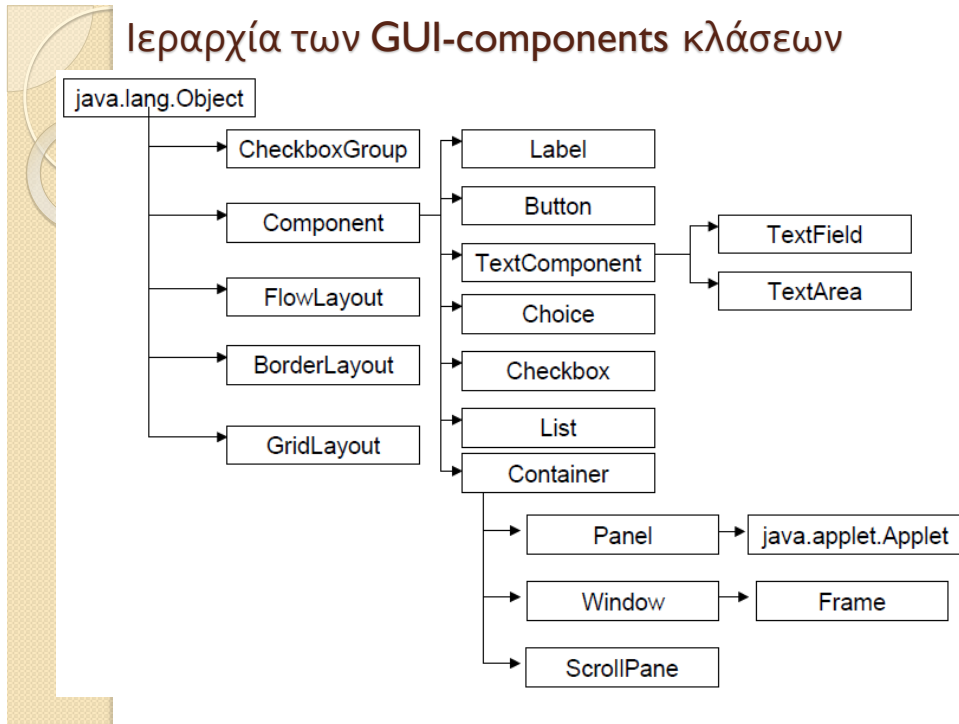
Οι κλάσεις GUI που χρησιμοποιούνται περισσότερο (1/2)

Label	Εμφανίζει κείμενο που δεν μπορεί να τροποποιηθεί από τον χρήστη (ετικέττα)
Button	Περιοχή που προξενεί ένα γεγονός (event) όταν επιλέγεται με το ποντίκι
TextField	Περιοχή όπου ο χρήστης εισάγει δεδομένα από το πληκτρολόγιο. Σε ένα TextField μπορούμε επίσης να εμφανίζουμε πληροφορίες
TextArea	Περιοχή όπου ο χρήστης εισάγει δεδομένα από το πληκτρολόγιο. Σε ένα TextArea μπορούμε να έχουμε πολλές γραμμές κειμένου σε αντίθεση με το TextField που μπορούμε να έχουμε μόνο μία.
Choice	Μια λίστα στοιχείων από τα οποία ο χρήστης μπορεί να επιλέξει ένα από αυτά με το ποντίκι.
Checkbox	Ένα boolean component που είναι επιλεγμένο ή μη επιλεγμένο. Με αυτό υλοποιούνται και τα Radio buttons (αμοιβαίως αποκλειόμενες επιλογές)

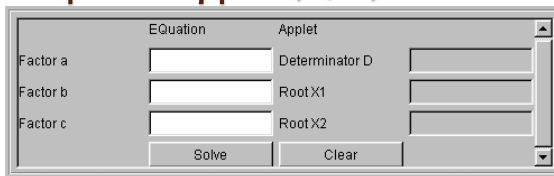
Οι κλάσεις GUI που χρησιμοποιούνται περισσότερο (2/2)

List	Λίστα στοιχείων από τα οποία ο χρήστης μπορεί να επιλέξει ένα από αυτά με το ποντίκι. Διπλό κλικ σε ένα στοιχείο της λίστας προξενεί ένα action event.
Panel	Είναι ένα container αντικείμενο στο οποίο μπορούν να τοποθετηθούν component αντικείμενα.
ScrollPane	Είναι ένα container αντικείμενο στο οποίο μπορεί να τοποθετηθεί ένα component, συνήθως ένα Panel, το οποίο θα εμφανίζεται στον χρήστη αλλά κι όταν αυτό δεν είναι εφικτό θα εμφανίζονται Scrollbars που θα επιτρέψουν την διολίσθηση του component έτσι ώστε να γίνεται ορατό και το τμήμα του που δεν φαίνεται.

Ιεραρχία των GUI-components κλάσεων



Παράδειγμα (1/2)



- Κατ' αρχήν γράφουμε τον κώδικα της μεθόδου `init` του applet ο οποίος θα δημιουργεί τα αναγκαία GUI components (TextField, Label, Button, Panel και ScrollPane).

```

// Equation applet
import java.applet.*;
import java.awt.*;
import java.awt.event.*;

public class eq extends java.applet.Applet { // Δηλώσεις των GUI components
    TextField tfA, tfB, tfC, tfD, tfX1, tfX2;
    Button btSolve, btClear;
    Panel p;
    ScrollPane sp;
}
  
```

Παράδειγμα (2/2)

```

public void init() { // Δημιουργία των GUI components.
//Δεν είναι ακόμη ορατά.
    p = new Panel();
    sp = new ScrollPane(ScrollPane.SCROLLBARS_AS_NEEDED);
    tfA = new TextField();
    tfB = new TextField();
    tfC = new TextField();
    tfD = new TextField();
    tfX1 = new TextField();
    tfX2 = new TextField();
    btSolve = new Button("Solve");
    btClear = new Button("Clear");
    tfD.setEditable(false);
    tfX1.setEditable(false);
    tfX2.setEditable(false);
}
}

```

Δομή ενός GUI

- Σύνθετα GUIs χωρίζονται σε υποπεριοχές (πχ αν θεωρήσουμε το πρόγραμμα Paint των Windows - το GUI του διαιρείται στην περιοχή των μενού επιλογών, στην περιοχή εργασίας, στην περιοχή παλέτας χρωμάτων κλπ)
- Για να απλοποιηθεί η ανάπτυξη ενός GUI στην Java, χωρίζεται συνήθως σε πολλά Panel components. Αξίζει να αναφερθεί ότι η κλάση Panel κληρονομεί την κλάση Container και η κλάση Applet την κλάση Panel. Έτσι τα Panels και τα Applets είναι Containers και μπορούν να έχουν Components, ακόμα και τύπου Panel.
- Ο κατασκευαστής της κλάσης Panel δεν παίρνει παραμέτρους. Τελικά προκύπτει μια δενδρική δόμηση του GUI που λέγεται component hierarchy ή containment hierarchy.
- Σε κάθε Panel τα Components που περιέχει είναι τοποθετημένα με ένα συγκεκριμένο πλάνο ή σχέδιο (layout). Η Java παρέχει διαχειριστές πλάνων (Layout Managers) για την διευθέτηση των Component αντικειμένων μέσα σε ένα Applet ή Panel.

Συνηθισμένες κλάσεις των Layout Managers

FlowLayout	Είναι ο default Layout Manager για τα Applets και τα Panels. Τοποθετεί τα Component αντικείμενα από αριστερά προς τα δεξιά με την σειρά που προστίθενται
BorderLayout	Τοποθετεί τα Component αντικείμενα σε πέντε περιοχές: Βόρεια, Νότια, Ανατολική, Δυτική και Κεντρική (North, South, East, West, Center).
GridLayout	Τοποθετεί τα Component αντικείμενα σε γραμμές και στήλες με καθορισμένη σειρά (πρώτα γεμίζει η πρώτη γραμμή μετά η δεύτερη γραμμή κλπ).
CardLayout	Τοποθετεί τα Component αντικείμενα σε στοίβα. Κάθε Container αντικείμενο στη στοίβα μπορεί να χρησιμοποιήσει οποιοδήποτε Layout Manager. Μόνο το Container αντικείμενο που βρίσκεται στην κορυφή της στοίβας είναι ορατό.
GridBagLayout	Παρόμοιος με τον GridLayout Manager. Διαφέρει στο ότι κάθε Component που προστίθεται μπορεί να έχει οποιοδήποτε μέγεθος (δηλαδή να καλύπτει περισσότερες από μια γραμμές και στήλες). Επίσης η σειρά με την οποία προστίθενται τα Component αντικείμενα στον GridBagLayout Manager μπορεί να είναι οποιαδήποτε.

Παράδειγμα Layout Managers (1/2)

// Τοποθέτηση των απλών GUI components μέσα σε ένα Panel.

// Γραμμή 1

```
p.setLayout(new GridLayout(5,4, 5,5));
```

```
p.add(new Label(""));
```

```
p.add(new Label("Equation"));
```

```
p.add(new Label("Applet"));
```

```
p.add(new Label(""));
```

// Γραμμή 2

```
p.add(new Label("Factor a"));
```

```
p.add(tfA);
```

```
p.add(new Label("Determinator D"));
```

```
p.add(tfD);
```

// Γραμμή 3

```
p.add(new Label("Factor b"));
```

Παράδειγμα Layout Managers (2/2)

```

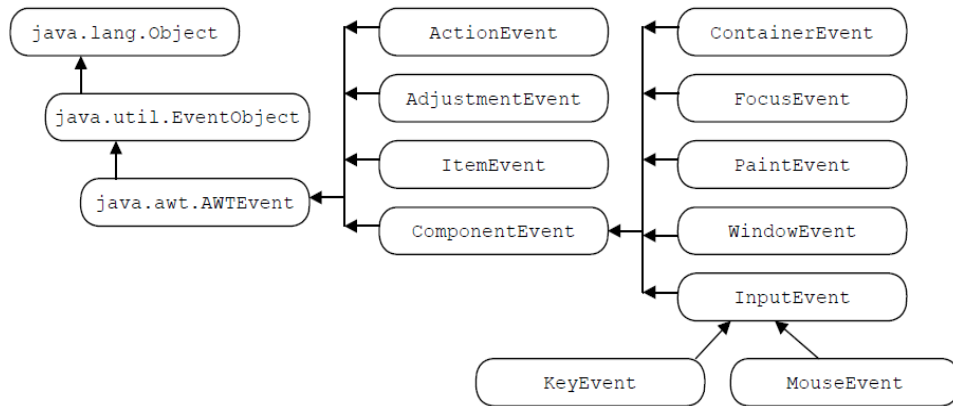
p.add(tfB); //hierarchy.
p.add(new Label("Root X1")); sp.add(p);
p.add(tfX1); sp.doLayout();
// Γραμμή 4 setLayout(new BorderLayout());
p.add(new Label("Factor c")); add(sp, BorderLayout.CENTER);
p.add(tfC);
p.add(new Label("Root X2"));
p.add(tfX2);
// Γραμμή 5
p.add(new Label(""));
p.add(btSolve);
p.add(btClear);
p.add(new Label(""));
// Δημιουργία του component

```

Γεγονότα στα GUI (GUI Events)

- Γεγονότα στέλνονται σε ένα πρόγραμμα Java από τα Windows.
- Η πληροφορία σχετικά με ένα GUI event, αποθηκεύεται σε ένα αντικείμενο που ανήκει στην κλάση ή σε υποκλάσεις της **AWTEvent**.
- Η AWTEvent είναι ορισμένη στο πακέτο java.awt.event. Το επόμενο σχήμα εμφανίζει την ιεραρχία των κλάσεων του πακέτου java.awt.event.

Ιεραρχία κλάσεων java.awt.event



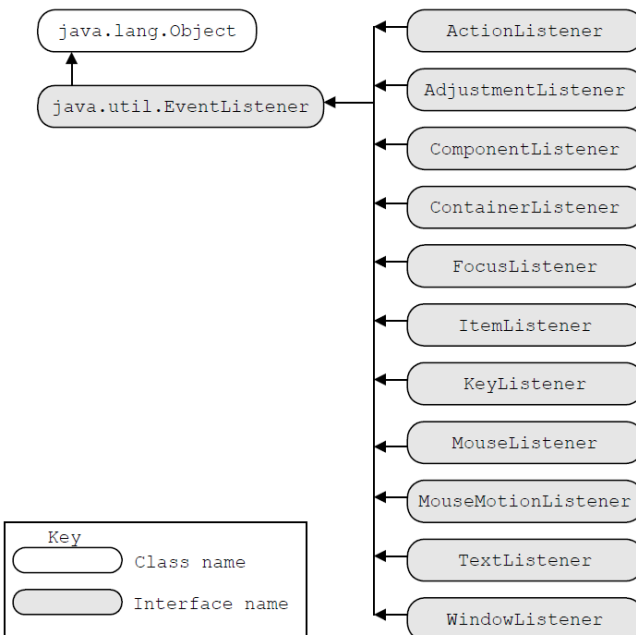
Πως μπορούμε να επεξεργαστούμε κάποιο event

- Για να μπορέσει ο προγραμματιστής να επεξεργαστεί κάποιο event το πετυχαίνει με δυο βήματα:
 - να δηλώσει έναν ακροατή γεγονότων (event listener) που θα ακούει τέτοιου τύπου events.
 - να υλοποιήσει τον χειριστή γεγονότων (event handler) που θα πιάνει events που προκύπτουν από το βήμα Α.

Event Listeners

- Ένας event listener πρέπει να είναι αντικείμενο κάποιας κλάσης, η οποία υλοποιεί ένα ή περισσότερα event-listener interface του πακέτου `java.awt.event`.
- Το επόμενο σχήμα εμφανίζει την ιεραρχία των event-listener interfaces του πακέτου `java.awt.event`.

Ιεραρχία των Event Listeners



Εξήγηση για τους Event Listeners

- Κάθε event-listener interface παρέχει μια ή περισσότερες event handler μεθόδους. Μια event handler μέθοδος καλείται αυτόματα όταν συμβαίνει κάποιο συγκεκριμένου τύπου γεγονός.
- Στην κλάση του event listener (είναι αυτονόητο ότι) θα πρέπει να υλοποιούνται όλες οι event handler μέθοδοι των event-listener interfaces που η κλάση αυτή υλοποιεί.
- Έτσι το event listener αντικείμενο της κλάσης αυτής, είναι σε θέση να ακούει συγκεκριμένου τύπου events που παράγονται από GUI-Components του προγράμματος - και σε περίπτωση που ένα τέτοιο event παραχθεί, εκτελείται η κατάλληλη event handler μέθοδος για αυτό το event.

Παράδειγμα Listeners (1/3)

```
btSolve.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent ae) {
        try {
            double a,b,c,d,x1,x2,sqrtd;
            a=(new Double(tfA.getText())).doubleValue();
            b=(new Double(tfB.getText())).doubleValue();
            c=(new Double(tfC.getText())).doubleValue();
            d = b*b-4*a*c;
            tfD.setText(Double.toString(d));
            if (d>0) {
                sqrtd = Math.sqrt(d);
                x1 = (-b-sqrtd)/(2*d*a);
                x2 = (-b+sqrtd)/(2*d*a);
            }
        }
    }
});
```

Παράδειγμα Listeners (2/3)

```

tfX1.setText(Double.toString(x1));
tfX2.setText(Double.toString(x2));

} else if (d==0) {
    x1 = x2 = -b/(2d*a);
    tfX1.setText(Double.toString(x1));
    tfX2.setText(Double.toString(x2));
} else {
    tfX1.setText("Error");
    tfX2.setText("Negative D");
}
} catch (Exception e) {
    tfD.setText(e.getMessage());
}
}
}

```

Παράδειγμα Listeners (3/3)

```

btClear.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent ae) {
        tfA.setText("");
        tfB.setText("");
        tfC.setText("");
        tfD.setText("");
        tfX1.setText("");
        tfX2.setText("");
    }
}
}

```

Adapter Classes

- Πρόκειται για προϋπάρχουσες κλάσεις που περιέχουν όλες τις συναρτήσεις ενός event listener interface με κενό σώμα. Ο προγραμματιστής, αντί να ορίσει εκ του μηδενός μια κλάση ακροατή και να ορίσει όλες τις μεθόδους του αντίστοιχου interface, απλά παρακάμπτει και χρησιμοποιεί μόνο τις μεθόδους της adapter κλάσης που τον ενδιαφέρουν. Είναι προφανές ότι αν κάποιος event listener interface περιέχει μία μόνο μέθοδο προς υλοποίηση τότε δεν ορίζεται για αυτό αντίστοιχη adapter κλάση.
- Παραδείγματα adapter κλάσεων είναι: η `MouseListenerAdapter` που υλοποιεί το interface `MouseListener`, η `MouseMotionAdapter` που υλοποιεί το interface `MouseMotionListener`, η `MouseMotionAdapter` που υλοποιεί το interface `MouseMotionAdapterListener` κλπ.

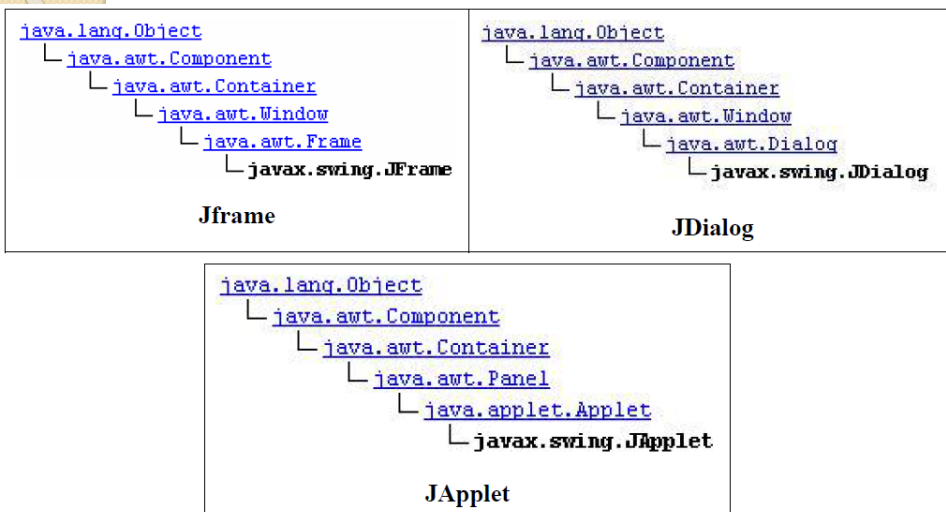
Γραφικές διεπαφές τύπου Swing

- Τα πακέτα που αρχίζουν με το πρόθεμα `javax.swing` παρέχουν ευέλικτα και ισχυρά εργαλεία ανάπτυξης GUI. Ιδιαίτερα το πακέτο `javax.swing` αναπτύχθηκε κυρίως λόγω των ανεπαρκειών του Abstract Windows Toolkit (AWT).
- Παραδείγματος χάριν, η κλάση `JButton` που ορίζεται σε αυτό, υπερτερεί της `AWT Button` κλάσης επιτρέποντας την ύπαρξη όχι μόνο απλού κείμενου, αλλά και εικόνων στα κουμπιά.

Top-Level Swing Containers και Swing Components

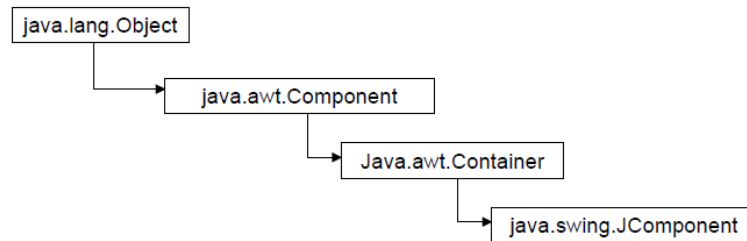
- Κάθε γραφική διεπαφή τύπου Swing πρέπει να έχει τουλάχιστον ένα top-level Swing container. Ένα top-level Swing container παρέχει την απαραίτητη υποστήριξη που χρειάζονται τα Swing components για την εμφάνισή τους και την διαχείριση των γεγονότων που αυτά παράγουν.
- Υπάρχουν τρία top-level Swing containers: το JFrame, το JDialog, και (για applets) το JApplet.
- Κάθε JFrame αντικείμενο δημιουργεί ένα κύριο γραφικό παράθυρο, κάθε JDialog αντικείμενο δημιουργεί ένα δευτερεύον παράθυρο (δηλ παράθυρο που εξαρτάται από κάποιο άλλο παράθυρο). Κάθε JApplet αντικείμενο δημιουργεί την περιοχή εμφάνισης ενός applet στο παράθυρο του Web Browser. Η ιεραρχία των top-level Swing containers δίνεται παρακάτω:

Η ιεραρχία των top-level Swing containers



Εξαιρέση

- Με εξαίρεση τα top-level containers, πχ το JFrame, όλα τα Swing components είναι υποκλάσεις της κλάσης JComponent. Η ιεραρχία της JComponent φαίνεται στο παρακάτω σχήμα:



Πίνακας εμφάνισης των swing Containers και Components

Top-Level Containers
Τα Container στην κορυφή οποιασδήποτε Swing ιεραρχίας.

The screenshot displays three distinct Swing components within a single window titled "Top-Level Containers". On the left is an "Applet" represented by a red apple icon. In the center is a "Dialog" box titled "An Inane Question" with the text "Would you like green eggs and ham?" and "Yes" and "No" buttons. On the right is a "Frame" window titled "FrameDemo" with standard window controls (minimize, maximize, close).

Πίνακας εμφάνισης των swing Containers και Components

A Label on a Panel
Color and font test:

- red
- blue
- green
- small

Panel

Scroll pane

Split pane

Tabbed pane

Tool bar

Πίνακας εμφάνισης των swing Containers και Components

Internal frame

Layered pane

Diagram illustrating the Swing window architecture:

- Frame
- Layered Pane
- Root pane
- Content Pane
- Menu Bar
- Glass Pane

Πίνακας εμφάνισης των swing Containers και Components

Basic Controls
Ατομικά Component που υπάρχουν προτίστως για να πάρουν δεδομένα από το χρήστη, επίσης εν γένει παρουσιάζουν κάποια κατάσταση.

The image displays several Swing components:

- Buttons:** A group of icons (pencil, eraser, highlighter, rubber) with a checked checkbox labeled "Check 1", a radio button labeled "Radio 2", and an "OK" button.
- Combo box:** A dropdown menu with "Pig" selected, and a list of other options: Bird, Cat, Dog, Rabbit, Pig.
- List:** A list box containing "January", "February", "March", and "April", with "March" selected.
- Menu:** A menu with two titles, "A Menu" and "Another Menu". Items include "A text-only menu item" (with Alt-1 shortcut), "Both text and icon" (with a sun icon), "A radio button menu item" (with a radio button), "A check box menu item" (with a checkbox), and "A submenu" (with a right-pointing arrow).
- Slider:** A slider labeled "Frames Per Second" with a range from 0 to 30 and a value of approximately 15.

Πίνακας εμφάνισης των swing Containers και Components

Uneditable Information Displays
Ατομικά Component που υπάρχουν απλώς για να δώσουν πληροφορίες στον χρήστη.

The image displays several Swing components:

- Spinner:** A small input field showing the number "20".
- Text field or Formatted text field:** A text input field with the text "Years: 30".
- Progress bar:** A horizontal progress bar showing 18% completion.
- Label:** A window titled "LabelDemo" containing an image of a cow and the text "Mooooooooo".

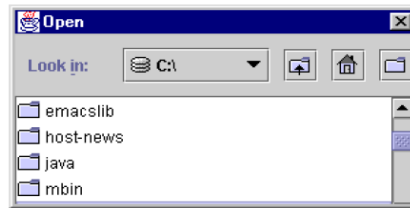
Πίνακας εμφάνισης των swing Containers και Components

Interactive Displays of Highly Formatted Information

Ατομικά Component που εμφανίζουν πληροφορίες με συγκεκριμένη μορφοποίηση η οποία (εάν επιλεγθούν) μπορεί να τροποποιηθεί από το χρήστη.



[Color chooser](#)



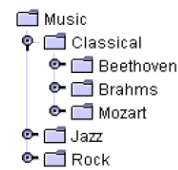
[File chooser](#)

First Name	Last Name	Favorite Food
Jeff	Dinkins	
Ewan	Dinkins	
Amy	Fowler	
Hania	Gajewska	
David	Geary	

[Table](#)



[Text](#)



[Tree](#)