



Ελληνική Δημοκρατία
Τεχνολογικό Εκπαιδευτικό
Ίδρυμα Ηπείρου

Προγραμματισμός Διαδικτύου

Ενότητα 5 : Γραφικά με την χρήση του πακέτου AWT

Ιωάννης Τσούλος



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Τμήμα Μηχανικών Πληροφορικής Τ.Ε

Προγραμματισμός Διαδικτύου

Ενότητα 5 : Γραφικά με την χρήση του πακέτου AWT

Ιωάννης Τσούλος

Επίκουρος Καθηγητής

Άρτα, 2015



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης





Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.





Χρηματοδότηση

- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «**Εκπαίδευση και Δια Βίου Μάθηση**» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο ΤΕΙ Ηπείρου**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



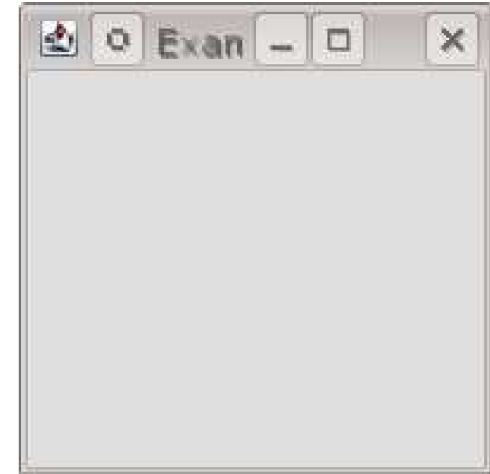
Δημιουργία απλών παραθύρων

- Για να δημιουργήσουμε ένα απλό παράθυρο θα πρέπει να χρησιμοποιήσουμε το αντικείμενο `Frame` του πακέτου `java.awt`.
- Αν δεν καλέσουμε την μέθοδο `show()` δεν θα εμφανιστεί το παράθυρο.
- Το παράθυρο που εμφανίζεται εδώ μπορεί να μεγαλώσει σε μέγεθος με την χρήση του ποντικιού.
- Αν δεν θέλουμε να γίνεται κάτι τέτοιο θα πρέπει να χρησιμοποιήσουμε την μέθοδο `setResizable(boolean)` περνώντας στην παράμετρο την τιμή `false` αν δεν επιτρέπουμε την αυξομείωση στο μέγεθος του παραθύρου και `true` σε άλλη περίπτωση.
- Από την άλλη αν θέλουμε να αλλάξουμε το μέγεθος του παραθύρου δεν έχουμε παρά να χρησιμοποιήσουμε την μέθοδο `resize(int,int)`.
- Η πρώτη παράμετρος είναι το μήκος του παραθύρου και η δεύτερη παράμετρος είναι το ύψος του παραθύρου.



Δημιουργία απλών παραθύρων

```
import java.awt.*;
public class Example1
{
    public static void main ( String arg[] )
    {
        Frame win=new Frame ( "Example1 ");
        win.resize(200,200);
        win.show();
    }
}
```





Δημιουργία απλών παραθύρων

■ Χρώματα (Color)

- `Color(int red,int green,int blue)` Δημιουργεί ένα χρώμα με τις αποχρώσεις `red`, `green` και `blue`.
- `Color(int rgb)` Δημιουργεί ένα χρώμα με τις αποχρώσεις να είναι κωδικοποιημένες στην μεταβλητή `rgb`. Η κωδικοποίηση γίνεται ως ακολούθως στα bits 16-23 μπαίνουν οι τιμές για κόκκινο στα bits 8-15 μπαίνουν οι τιμές για το πράσινο και στα bits 0-7 οι τιμές για το γαλάζιο.
- `int getRed ()` Επιστρέφει την απόχρωση του κόκκινου για το χρώμα.
- `int getGreen()` Επιστρέφει την απόχρωση του πράσινου για το χρώμα.
- `int getBlue ()` Επιστρέφει την απόχρωση του μπλε για το χρώμα.

Table 1: Οι βασικοί χρωματικοί συνδυασμοί.

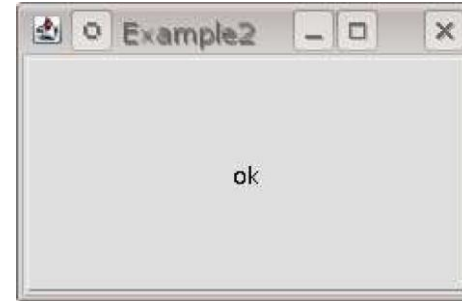
METABΛΗΤΗ	KOKKINO	ΠΡΑΣΙΝΟ	ΜΠΛΕ
white	255	255	255
lightGray	192	192	192
gray	128	128	128
darkGray	64	64	64
black	0	0	0
red	255	0	0
pink	255	175	175
orange	255	200	0
yellow	255	255	0
green	0	255	0
magenta	255	0	255
cyan	0	255	255
blue	0	0	255



Πλήκτρα πιέσεως - Σχεδιασμός

```
import java.awt.*;
class MyFrame extends Frame
{
    Button ok;
    MyFrame ( String title)
    {
        super(title );
        resize(100,100);
        ok=new Button ( "ok" );
        add(ok);
    }
}

public class Example2
{
    public static void main) String arg[])
    {
        MyFrame win=new MyFrame ("Example2");
        win.show();
    }
}
```





Πλήκτρα πιέσεως - Σχεδιασμός

```
import java.awt.*;
class MyFrame extends Frame
{
    Button ok;
    MyFrame (String title)
    {
        super(title );
        resize (100,100):
        setLayout (new FlowLayout() );
        ok=new Button ( "ok" );
        add(ok):
    }
}

public class Example3
{
    public static void main ( String arg[] )
    {
        MyFrame win=new MyFrame ( "Example3" );
        win.resize (200,200); win.show():
    }
}
```

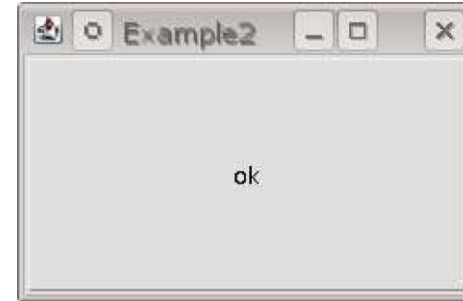




Γεγονότα

```
import java.awt.*;
class MyFrame extends Frame
{
    Button ok;
    MyFrame ( String title)
    {
        super(title );
        resize(100,100);
        ok=new Button ( "ok" );
        add(ok);
    }
}

public class Example2
{
    public static void main) String arg[])
    {
        MyFrame win=new MyFrame ("Example2");
        win.show();
    }
}
```





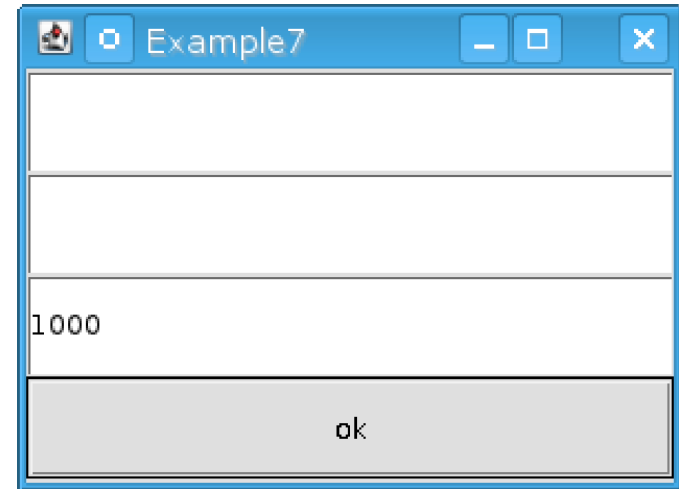
Γεγονότα

1. **keyDown(Event evt,int key)** Ενεργοποιείται αν πατηθεί το πλήκτρο key του πληκτρολογίου.
2. **keyUp(Event evt,int key)** Ενεργοποιείται αν ελευθερωθεί το πλήκτρο key του πληκτρολογίου.
3. **lostFocus(Event evt,Object obj)** Ενεργοποιείται αν αφαιρεθεί η εστίαση από το συγκεκριμένο παράθυρο Η εστίαση δεν έχει να κάνει σε κάτι με την java αλλά με το σύστημα παραθύρων που χρησιμοποιείται.
4. **gotFocus(Event evt,Object obj)** Ενεργοποιείται όταν το παράθυρο της εφαρμογής μας αποκτήσει εστίαση.
5. **mouseDown(Event evt,int x,int y)** Ενεργοποιείται όταν πατηθεί κάποιο πλήκτρο του ποντικιού στην θέση (x,y) του παραθύρου εφαρμογής.
6. **mouseUp(Event evt,int x,int y)** Ενεργοποιείται όταν ελευθερωθεί κάποιο πλήκτρο του ποντικιού στην θέση (x,y) του παραθύρου εφαρμογής.
7. **mouseMove(Event evt,int x,int y)** Ενεργοποιείται όταν κινηθεί το ποντικιού στην θέση (x,y) του παραθύρου εφαρμογής.
8. **mouseDrag(Event evt,int x,int y)** Ενεργοποιείται όταν έχουμε σύρσιμο με το ποντίκι στην θέση (x,y) του παραθύρου εφαρμογής.
9. **mouseenter(Event evt,int x,int y)** Ενεργοποιείται όταν έχουμε είσοδο στο παράθυρο εφαρμογής στην θέση (x,y)
10. **mouseExit(Event evt,int x,int y)** Ενεργοποιείται όταν φεύγει το ποντίκι από το παράθυρο εφαρμογής.



Γεγονότα

```
import java.awt.*;
class MyFrame extends Frame
{
    Button ok, cancel, clear;
    MyFrame(String title)
    {
        super(title);
        resize(150,50);
        setLayout(new GridLayout(1,3));
        ok=new Button("ok");
        cancel=new Button("cancel");
        clear=new Button("clear");
        add(ok); add(cancel); add(clear);
    }
    public boolean action(Event evt, Object arg)
    {
        if(evt.target.equals(ok))
            System.out.println("ok_pressed");
        else
            if(evt.target.equals(cancel))
                System.out.println("cancel_pressed");
            else
                if(evt.target.equals(clear))
                    System.out.println("clear_pressed");
                else
                    super.action(evt, arg);
        return true;
    }
}
public class Example6
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("Example6");
        win.show();
    }
}
```





Πλαίσια κειμένου

■ Απλά πλαίσια

- Τα απλά πλαίσια είναι αντικείμενα της κατηγορίας `TextField`.
- Η συνάρτηση δημιουργίας δέχεται δύο ορίσματα: το εξ ορισμού κείμενο και το μέγιστο πλήθος γραμμμάτων στο κείμενο
- Η μέθοδος `parseDouble()` της κατηγορίας `Double` μετατρέπει (αν γίνεται) το όρισμα που της δίνεται σαν αλφαριθμητικό σε αριθμό κινητής υποδιαστολής.
- Όμοια υπάρχει και η μέθοδος `parseInt()` της κατηγορίας `Integer`.

■ Πλαίσια πολλών γραμμών

- Τα πλαίσια μίας γραμμής είναι καλά για την περίπτωση που έχουμε πληροφορία η οποία μπορεί να συμπυκνωθεί σε μία γραμμή.
- Για την περίπτωση που έχουμε κείμενο πολλών γραμμών θα πρέπει να χρησιμοποιήσουμε την κατηγορία `TextArea`.
- Οι παράμετροι της μεθόδου δημιουργίας είναι το προκαθορισμένο κείμενο το πλήθος των γραμμών και το πλήθος των στηλών της περιοχής.



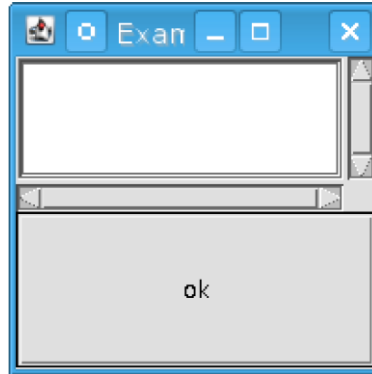
Πλαίσια κειμένου

Μια πρώτη χρήση των απλών πλαισίων

```
import java.awt.*;
class MyFrame extends Frame
{
    Button ok;
    TextField name,lastname, salary;
    MyFrame(String title)
    {
        super(title);
        resize(300,220);
        setLayout(new GridLayout(4,1));
        ok=new Button("ok");
        name=new TextField("",30);
        lastname=new TextField("",30);
        salary=new TextField("1000",30);
        add(name);
        add(lastname);
        add(salary);
        add(ok);
    }
    public boolean action(Event evt, Object arg)
    {
        Object comp=evt.target;
        String text;
        double Salary;
        if(comp.equals(ok))
        {
            text=name.getText();
            System.out.println("YOUR_NAME="+text);
            text=lastname.getText();
            System.out.println("YOUR_LASTNAME="+text);
            text=salary.getText();
            Salary=Double.parseDouble(text);
            System.out.println("YOUR_SALARY="+Salary);
            System.exit(0);
        }
        else
            super.action(evt, arg);
        return true;
    }
}
public class Example7
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("Example7");
        win.show();
    }
}
```

Παράδειγμα χρήσεως πλαισίου πολλών γραμμών και στηλών

```
import java.awt.*;
class MyFrame extends Frame
{
    TextArea area;
    Button ok;
    MyFrame(String title)
    {
        super(title);
        resize(200,200);
        setLayout(new GridLayout(2,1));
        area=new TextArea("",10,10);
        ok=new Button("ok");
        add(area);
        add(ok);
    }
    public boolean action(Event evt, Object arg)
    {
        if(evt.target.equals(ok))
        {
            System.out.println("TEXT="+area.getText());
            System.exit(0);
        }
        else
            super.action(evt, arg);
        return true;
    }
}
public class Example8
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("Example8");
        win.show();
    }
}
```





Μενού

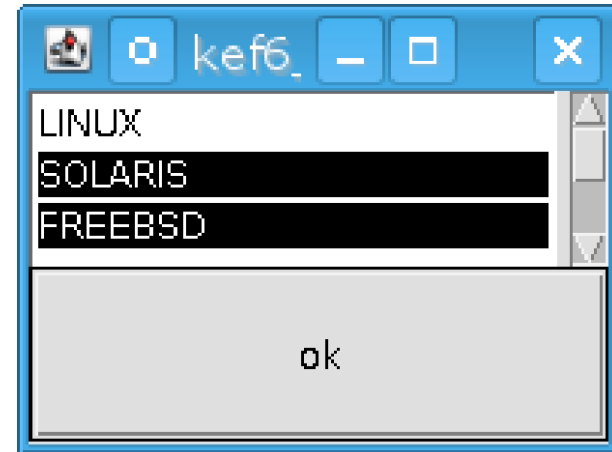
■ Λίστες επιλογής

- Οι λίστες επιλογής είναι οπτικά συστατικά στα οποία μία σειρά από περιπτώσεις εμφανίζονται σχεδόν πάντα όλες στην οθόνη και μπορούμε να διαλέξουμε μία ή περισσότερες από αυτές τις περιπτώσεις.
- Η κατηγορία που υλοποιεί αυτό το οπτικό συστατικό είναι η List.
- Η συνάρτηση δημιουργίας δέχεται δύο ορίσματα: το πλήθος των ορισμάτων που θα φαίνονται (ακέραιος αριθμός) και μία λογική τιμή που καθορίζει αν επιτρέπουμε πολλαπλή επιλογή
- Η μέθοδος `getSelectedItems()` επιστρέφει σε πίνακα από αλφαριθμητικά τα επιλεγμένα στοιχεία της λίστας



Μενού

```
import java.awt.*;
class MyFrame extends Frame
{
    List list;
    Button ok;
    MyFrame(String title)
    {
        super(title);
        resize(200,150);
        setLayout(new GridLayout(2,1));
        list=new List(4,true);
        list.addItem("LINUX");
        list.addItem("SOLARIS");
        list.addItem("FREEBSD");
        list.addItem("OPENBSD");
        list.addItem("BEOS");
        list.addItem("WIN98");
        ok=new Button("ok");
        add(list);
        add(ok);
    }
    public boolean action(Event evt, Object arg)
    {
        if(evt.target.equals(ok))
        {
            String items[]=list.getSelectedItems();
            for(int i=0;i<items.length;i++)
            {
                System.out.println("SELECTED="+items[i]);
            }
            System.exit(0);
        }
        return true;
    }
}
public class Example9
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("kef6_9");
        win.show();
    }
}
```





Μενού

■ Choice

- Τα choice boxes είναι ένα άλλο είδος λίστας διαφορετικό από αυτό της λίστας.
- Η διαφορά έγκειται στο ότι τα στοιχεία που μπορούμε να επιλέξουμε δεν φαίνονται στην οθόνη μας.
- Η κατηγορία ονομάζεται Choice και η συνάρτηση δημιουργίας δεν παίρνει κανένα όρισμα.

■ Menubar

- Με τον όρο Menubar εννοούμε την γραμμή μενού που εμφανίζεται στην κορυφή μίας εφαρμογής κάτι που είναι ιδιαίτερα συνηθισμένο στα προγράμματα που διαθέτουν γραφική διαπροσωπεία.



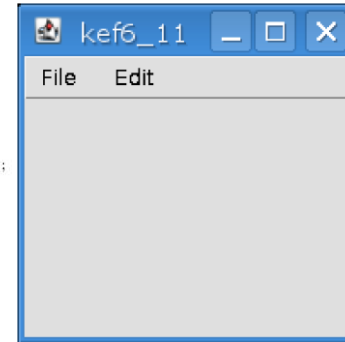
Μενού

Παράδειγμα χρήσεως Choice

```
import java.awt.*;
class MyFrame extends Frame
{
    Choice list;
    Button ok;
    MyFrame(String title)
    {
        super(title);
        resize(200,150);
        setLayout(new GridLayout(2,1));
        list=new Choice();
        list.addItem("LINUX");
        list.addItem("SOLARIS");
        list.addItem("FREEBSD");
        list.addItem("OPENBSD");
        list.addItem("BEOS");
        list.addItem("WIN98");
        ok=new Button("ok");
        add(list);
        add(ok);
    }
    public boolean action(Event evt, Object arg)
    {
        if(evt.target.equals(ok))
        {
            String item=list.getSelectedItem();
            System.out.println("SELECTED="+item);
            System.exit(0);
        }
        else
            super.action(evt, arg);
        return true;
    }
}
public class Example10
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("kef6_10");
        win.show();
    }
}
```

Παράδειγμα χρήσεως MenuBar

```
import java.awt.*;
class MyFrame extends Frame
{
    MenuBar bar;
    Menu file, edit;
    MenuItem itemFile[], itemEdit[];
    MyFrame(String title)
    {
        super(title);
        resize(200,200);
        setLayout(new FlowLayout());
        itemFile=new MenuItem[3];
        itemEdit=new MenuItem[3];
        itemFile[0]=new MenuItem("New");
        itemFile[1]=new MenuItem("Save");
        itemFile[2]=new MenuItem("Exit");
        itemEdit[0]=new MenuItem("Cut");
        itemEdit[1]=new MenuItem("Copy");
        itemEdit[2]=new MenuItem("Paste");
        file=new Menu("File");
        edit=new Menu("Edit");
        for(int i=0;i<3;i++)
        {
            file.add(itemFile[i]);
            edit.add(itemEdit[i]);
        }
        bar=new MenuBar();
        bar.add(file);
        bar.add(edit);
        setMenuBar(bar);
    }
    public boolean action(Event evt, Object arg)
    {
        if(evt.target instanceof MenuItem)
        {
            String text=(String)arg;
            System.out.println("Selected _="+arg);
            if(text.equals("Exit"))
                System.exit(0);
        }
        else
            super.action(evt, arg);
        return true;
    }
}
public class Example11
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("kef6_11");
        win.show();
    }
}
```



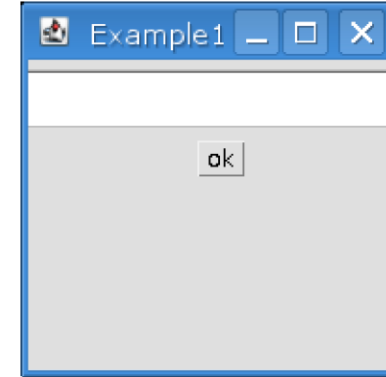


Μενού

■ Στατικό κείμενο

- Στατικό κείμενο είναι οι ετικέτες που εμφανίζονται στο κείμενο.
- Πολλές φορές τις χρησιμοποιούμε για να δώσουμε πληροφορία για κάποιο οπτικό συστατικό ή και να πληροφορήσουμε τον χρήστη για κάποια αλλαγή.
- Για να δημιουργήσουμε ένα αντικείμενο της κατηγορίας βάζουμε το κείμενο που θέλουμε στην μέθοδο δημιουργίας της κατηγορίας.

```
import java.awt.*;
class MyFrame extends Frame
{
    TextField text;
    Label length;
    Button ok;
    MyFrame(String title)
    {
        super(title);
        resize(200,200);
        setLayout(new FlowLayout());
        ok=new Button("ok");
        text=new TextField("",30);
        length=new Label("");
        add(text);
        add(length);
        add(ok);
    }
    public boolean keyDown(Event evt,int key)
    {
        String Text=text.getText();
        String lbl="" +Text.length();
        length.setText(lbl);
    }
    return true;
}
public boolean action(Event evt, Object arg)
{
    if(evt.target.equals(ok))
    {
        System.exit(0);
    }
    return true;
}
}
public class Example12
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("Example12");
        win.show();
    }
}
```





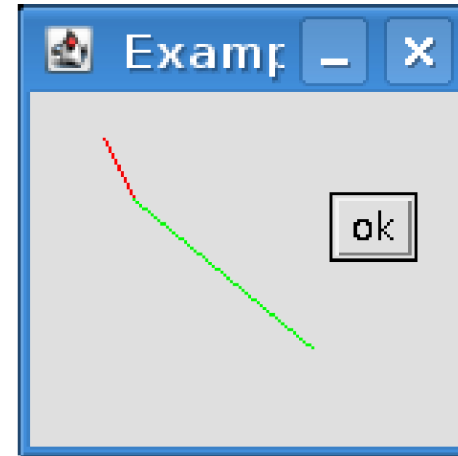
Γραφικά δύο διαστάσεων

- Για να σχεδιάσουμε απλά γραφικά πρέπει να χρησιμοποιήσουμε αντικείμενα της κατηγορίας Graphics.
- **Γραμμές**
 - Για να σχεδιάσουμε απλές γραμμές πρέπει να χρησιμοποιήσουμε την μέθοδο `drawLine(int, int, int, int)` της κατηγορίας Graphics.
 - Οι δύο πρώτες παράμετροι είναι το αρχικό σημείο της ευθείας και οι δύο επόμενες το τελικό σημείο στην ευθεία.
- **Ορθογώνια**
 - Για τον σχεδιασμό ορθογωνίων διατίθενται μία σειρά από χρήσιμες συναρτήσεις
 - `void drawRect(int x, int y, int width, int height)`: Ορθογώνιο με πάνω άκρο στο (x,y) με πλάτος `width` και ύψος `height`.
 - `void fillRect(int x, int y, int width, int height)`: Γεμάτο ορθογώνιο με πάνω άκρο στο (x,y) με πλάτος `width` και ύψος `height`.
 - `void draw3DRect(int x, int y, int width, int height, boolean flag)`: Ορθογώνιο με πάνω άκρο στο (x,y) με πλάτος `width` και ύψος `height`.



Γραφικά δύο διαστάσεων

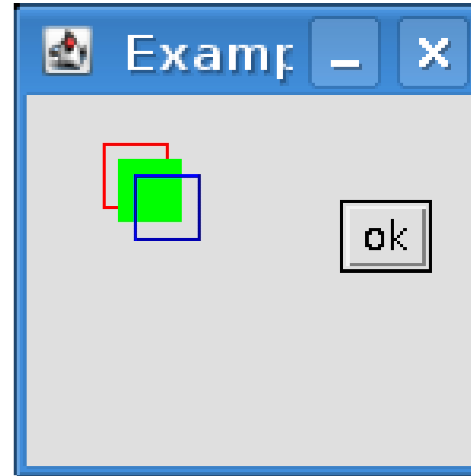
```
import java.awt.*;
class MyCanvas extends Canvas
{
    MyCanvas() {}
    public void paint(Graphics g)
    {
        g.setColor(new Color(255,0,0));
        g.drawLine(10,10,20,30);
        g.setColor(new Color(0,255,0));
        g.drawLine(20,30,80,80);
    }
}
class MyFrame extends Frame
{
    MyCanvas canvas;
    Button ok;
    MyFrame(String title)
    {
        super(title);
        resize(150,150);
        setLayout(new FlowLayout());
        ok=new Button("ok");
        canvas=new MyCanvas();
        canvas.resize(80,80);
        add(canvas);
        add(ok);
    }
    public boolean action(Event evt, Object arg)
    {
        if(evt.target.equals(ok))
            System.exit(0);
        else
            super.action(evt, arg);
        return true;
    }
}
public class Example13
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("Example13");
        win.show();
    }
}
```





Γραφικά δύο διαστάσεων

```
import java.awt.*;
class MyCanvas extends Canvas
{
    MyCanvas()
    {
    }
    public void paint(Graphics g)
    {
        g.setColor(new Color(255,0,0));
        g.drawRect(10,10,20,20);
        g.setColor(new Color(0,255,0));
        g.fillRect(15,15,20,20);
        g.setColor(new Color(0,0,255));
        g.draw3DRect(20,20,20,20,true);
    }
}
class MyFrame extends Frame
{
    MyCanvas canvas;
    Button ok;
    MyFrame(String title)
    {
        super(title);
        resize(150,150);
        setLayout(new BorderLayout());
        ok=new Button("ok");
        canvas=new MyCanvas();
        canvas.resize(80,80);
        add(canvas);
        add(ok);
    }
    public boolean action(Event evt, Object arg)
    {
        if(evt.target.equals(ok))
            System.exit(0);
        else
            super.action(evt, arg);
        return true;
    }
}
public class Example14
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("Example14");
        win.show();
    }
}
```





Γραφικά δύο διαστάσεων

▪ Ελλείψεις

- Για τον σχεδιασμό ελλείψεων από το πακέτο AWT διατίθενται μία σειρά από συναρτήσεις
 - `void drawOval(int x,int y,int width,int height)` Σχεδιάζει μία έλλειψη η οποία περικλείεται στο ορθογώνιο με πάνω αριστερό άκρο στο (x,y) με πλάτος width και με ύψος height.
 - `void fillOval(int x,int y,int width,int height)` Σχεδιάζει μία πλήρη έλλειψη η οποία περικλείεται στο ορθογώνιο με πάνω αριστερό άκρο στο (x,y) με πλάτος width και με ύψος height.
 - `void drawArc(int x,int y,int width,int height,int start,int length)` Σχεδιάζει ένα ελλειπτικό τόξο της ελλείψεως που περικλείεται στο ορθογώνιο με πάνω αριστερό άκρο στο (x,y) με πλάτος width και με ύψος height. Το τόξο ξεκινάει από την γωνία start και τελειώνει στην γωνία end.
 - `void fillArc(int x,int y,int width,int height,int start,int length)` Σχεδιάζει ένα γεμάτο ελλειπτικό τόξο της ελλείψεως που περικλείεται στο ορθογώνιο με πάνω αριστερό άκρο στο (x,y) με πλάτος width και με ύψος height. Το τόξο ξεκινάει από την γωνία start και τελειώνει στην γωνία end.



Γραφικά δύο διαστάσεων

▪ Κείμενο

- Για το σχεδιασμό κειμένου χρησιμοποιούμε την μέθοδο `drawString(String text, int x, int y)` η οποία σχεδιάζει στο σημείο (x,y) το αλφαριθμητικό `text`.
- Για να σχεδιάσουμε με την γραμματοσειρά της επιλογής μας πρέπει να καλέσουμε την μέθοδο `setFont (Font)` της κατηγορίας `Graphics`.
- Το αντικείμενο `Font` έχει τις ακόλουθες ενδιαφέρουσες μεθόδους
 - `Font(String name, int style, int size)`: Φτιάχνει μία γραμματοσειρά της οικογένειας `name` (πχ. `Helvetica`) με είδος `style` (`PLAIN`, `BOLD`, `ITALIC`) και με μέγεθος `size`.
 - `String getName()`: Επιστρέφει την οικογένεια της γραμματοσειράς.
 - `int getSize()`: Επιστρέφει το μέγεθος της γραμματοσειράς
 - `int getStyle()`: Επιστρέφει το είδος της γραμματοσειράς



Γραφικά δύο διαστάσεων

Παράδειγμα εμφάνιση κειμένου

```
import java.awt.*;
class MyCanvas extends Canvas
{
    MyCanvas()
    {
    }
    public void paint(Graphics g)
    {
        g.setFont(new Font("Helvetica",Font.ITALIC,10));
        g.drawString("GOOD_MORNING!",20,20);
    }
}
class MyFrame extends Frame
{
    MyCanvas canvas;
    Button ok;
    MyFrame(String title)
    {
        super(title);
        resize(250,100);
        setLayout(new FlowLayout());
        ok=new Button("ok");
        canvas=new MyCanvas();
        canvas.resize(180,80);
        add(canvas);
        add(ok);
    }
    public boolean action(Event evt, Object arg)
    {
        if(evt.target.equals(ok))
            System.exit(0);
        else
            super.action(evt, arg);
        return true;
    }
}
public class Example15
{
    public static void main(String arg[])
    {
        MyFrame win=new MyFrame("Example15");
        win.show();
    }
}
```



Βιβλιογραφία

- Πλήρες Εγχειρίδιο της Java 2 Platform, Laura Lemay & Rogers Cadenhead, Ελληνική έκδοση Γκιούρδα
- Java Προγραμματισμός, 6η έκδοση, Harvey M. Deitel, Paul J. Deitel . Εκδόσεις Μ. Γκιούρδας, Αθήνα 2005.
- Εισαγωγή στη γλώσσα προγραμματισμού Java, Εμμ. Σκορδαλάκης, Εργαστήριο Λογισμικού Ομάδα Τεχνολογίας Λογισμικού ΕΜΠ
- Java με UML, Else Lervik και Vegard B. Havdal, Εκδόσεις Κλειδάριθμος, 2005.
- Προγραμματισμός με Java, Γιάννη Κάβουρα, Εκδόσεις Κλειθάρθμος, Αθήνα 2003.
- Java in a nutshell, Deluxe Edition και Java Examples in a Nutshell, David Flanagan, Εκδόσεις OReilly
- Java Collections An Introduction to ADTs, Data Structures and Algorithms, David Watt, Deryck Brown, John Wiley and Sons
- Thinking in Java (3rd edition), B. Eckel, Pearson Prentice Hall, 2003.
- Introduction to Java Programming (5th edition), Y. Daniel Liang, Pearson Prentice Hall, 2005.



Σημείωμα Αναφοράς

Copyright Τεχνολογικό Ίδρυμα Ηπείρου. Ιωάννης Τσούλος.
Προγραμματισμός Διαδικτύου.

Έκδοση: 1.0 Άρτα, 2015. Διαθέσιμο από τη δικτυακή
διεύθυνση:

<http://eclass.teiep.gr/OpenClass/courses/COMP114/>



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά Δημιουργού-Μη Εμπορική Χρήση-Όχι Παράγωγα Έργα 4.0 Διεθνές [1] ή μεταγενέστερη. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, Διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/deed.el>



Τέλος Ενότητας

Επεξεργασία: Ευάγγελος Καρβούνης
Άρτα, 2015



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Τέλος Ενότητας

Γραφικά με την χρήση του πακέτου AWT



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

