



<b>Εξάμηνο</b>	2013 – 2014 Εαρινό
<b>Μάθημα</b>	Αντικειμενοστραφής Προγραμματισμός
<b>Τίτλος</b>	Applets

## A. Ύλη εργαστηρίου

- 6.1 Applets
- 6.2 Μέθοδοι start, init, paint, stop και destroy
- 6.3 Γραφικά
- 6.4 Χρώματα
- 6.5 Σχήματα

### Προετοιμασία εργαστηρίου:

- Στο φάκελο <Τα έγγραφα μου> ή <My Documents> δημιουργήστε φάκελο με τον αριθμό μητρώου σας π.χ **TXXX** και εκεί δημιουργήστε το φάκελο **Lab7**. **Όλες οι ασκήσεις που ακολουθούν για αυτό το εργαστήριο θα αποθηκευθούν εκεί και μόνον εκεί.** Όταν τελειώσει το εργαστήριο αυτό μπορείτε να αντιγράψετε το φάκελο για να έχετε τις ασκήσεις και να συνεχίσετε να εργάζεστε εκτός εργαστηρίου.

## B. Ασκήσεις λυμένες για το εργαστήριο

### Άσκηση 1<sup>η</sup>

Να γράψετε ένα applet ShowColors το οποίο να τυπώνει ένα μήνυμα με πολλούς συνδυασμούς χρωμάτων.

#### // ShowColors.java

```
import java.awt.*;
import java.applet.*; // εισαγωγή πακέτου για χρήση των applets

public class ShowColors extends Applet { //επεκτείνω την υπάρχουσα κλάση Applet
    // προσοχή το όνομα του πακέτου είναι applet ενώ
    // το όνομα της κλάσης που επεκτείνω είναι Applet

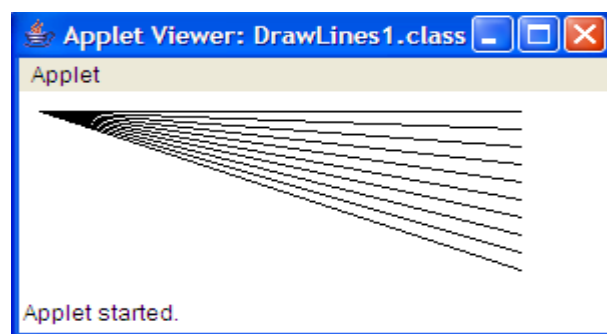
    public void init() {
    }

    public void paint(Graphics g) {
        for (int i=1; i<11;i++ ) {
            // set new drawing color using integers
            int rc = (int)(Math.random()*256);
            int gc = (int)(Math.random()*256);
            int bc = (int)(Math.random()*256);
        }
    }
}
```

```
        Color col = new Color(rc,gc,bc);
        g.setColor( col );
        g.drawString( "Hello ! Welcome to Java programming with
applets ", 130, 30*i );
    }
}
```

## Άσκηση 2<sup>η</sup>

Να γράψετε ένα applet DrawLines1 το οποίο να σχεδιάζει 10 ευθύγραμμα τμήματα όπως στο παρακάτω σχήμα.



```
import java.awt.*;
import java.applet.*;

public class DrawLines1 extends Applet {

    public void init() {
    }

    public void paint(Graphics g) {
        int counter = 1;
        while (counter<=10){
            g.drawLine(10,10,250, counter *10);
            ++counter;
        }
    }
}
```

## Άσκηση 3<sup>η</sup>

Να γράψετε ένα applet TestPolygon1 το οποίο να σχεδιάζει ένα τυχαίο πεντάγωνο.

```
import java.awt.*;
import java.applet.*;

public class TestPolygon1 extends Applet {

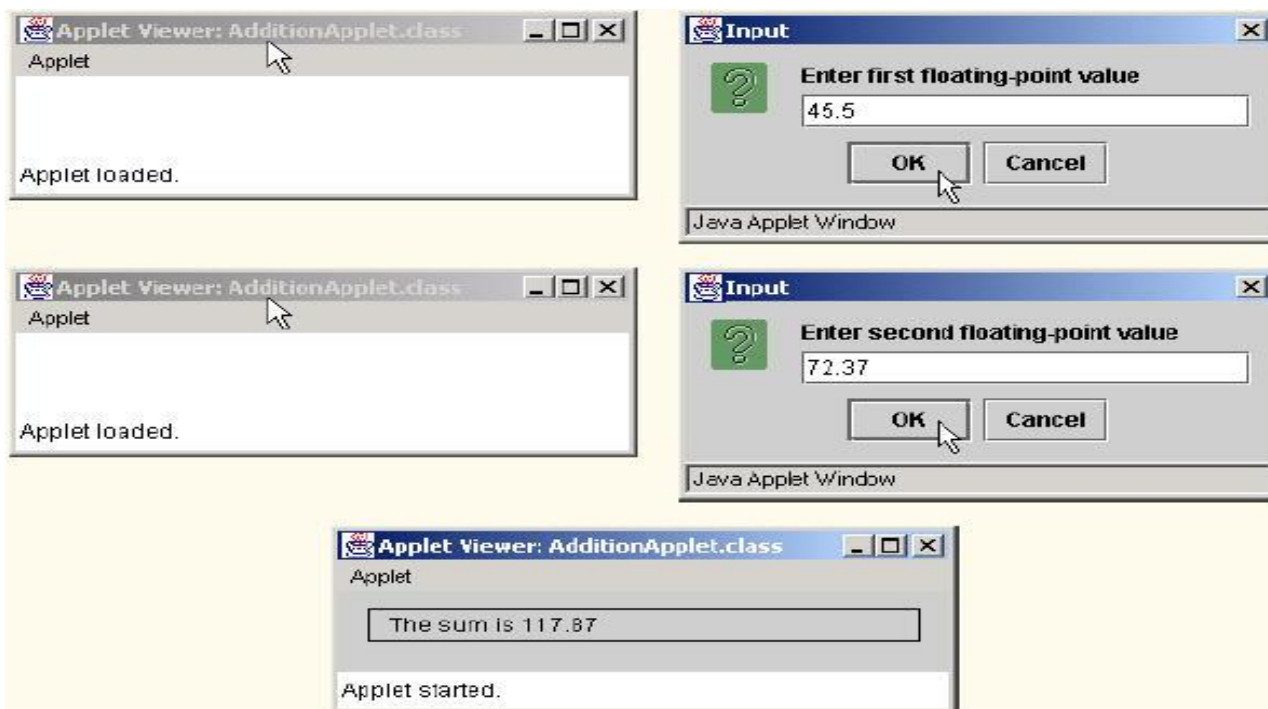
    public void init() {
    }
}
```

```
public void paint(Graphics g) {  
    int xpoints[]={30,140,160,130,50};  
    int ypoints[]={40,70,100,140,120};  
    int points=5;  
    g.drawPolygon(xpoints,ypoints,points);  
}  
}
```

#### Άσκηση 4η

Δημιουργήστε ένα applet που θα προσθέτει 2 Double precision αριθμούς. Αυτό το applet θα ζητά από το χρήστη να εισάγει δύο αριθμούς κινητής υποδιαστολής (δηλαδή, τους αριθμούς με δεκαδικό σημείο, όπως 7.33, 0.0975 και 1000.12345). Για την αποθήκευση αριθμών κινητής υποδιαστολής στη μνήμη μας εισαγάγει primitive τύπου δεδομένων διπλό(double), το οποίο αντιπροσωπεύει διπλής ακρίβειας αριθμών κινητής υποδιαστολής(double-precision floating-point). Υπάρχει επίσης τύπος δεδομένων float για την αποθήκευση και μονής ακρίβειας αριθμών (single-precision) κινητής υποδιαστολής. Μια διπλή απαιτεί περισσότερη μνήμη για την αποθήκευση μιας κινητής υποδιαστολής τιμή, αλλά το αποθηκεύει με περίπου δύο φορές την ακρίβεια ενός float (15 ψηφία για το σημαντικό διπλό έναντι επτά σημαντικά ψηφία για float). Για μία ακόμη φορά, να ζητήσει JOptionPane.showInputDialog χρήση εισροών(input) από το χρήστη. Η εφαρμογή υπολογίζει το άθροισμα των τιμών των εισροών και εμφανίζει το αποτέλεσμα με βάση μια σειρά μέσα σε ένα ορθογώνιο, για την εφαρμογή. Το αρχείο HTML για να φορτώσει αυτό το applet σε appletviewer είναι το παρακάτω:

```
1 <html>  
2 <applet code = "AdditionApplet.class" width = "300" height = "65">  
3 </applet>  
4 </html>
```



```
/*
 * AdditionApplet.java
 *
 *
 * To change this template, choose Tools | Template Manager
 * and open the template in the editor.
 */

// Adding two floating-point numbers.

// Java core packages
import java.awt.Graphics; // import class Graphics

// Java extension packages
import javax.swing.*; // import package javax.swing

public class AdditionApplet extends JApplet
{
    double sum; // sum of values entered by user
    // initialize applet by obtaining values from user
    public void init()
    {
        String firstNumber; // first string entered by user
        String secondNumber; // second string entered by user
        double number1; // first number to add
        double number2; // second number to add

        // obtain first number from user
        firstNumber = JOptionPane.showInputDialog( "Enter first floating-
        point value" );

        // obtain second number from user
```

```
secondNumber = JOptionPane.showInputDialog(
    "Enter second floating-point value" );

    // convert numbers from type String to type double
    number1 = Double.parseDouble( firstNumber );
    number2 = Double.parseDouble( secondNumber );

    // add numbers
    sum = number1 + number2;
}

// draw results in a rectangle on applet's background
public void paint( Graphics g )
{
    // call inherited version of method paint
    super.paint( g );

    // draw rectangle starting from (15, 10) that is 270
    // pixels wide and 20 pixels tall
    g.drawRect( 15, 10, 270, 20 );

    // draw results as a String at (25, 25)
    g.drawString( "The sum is " + sum, 25, 25 );

} // end method paint
} // end class AdditionApplet
```

## Άσκηση 5η

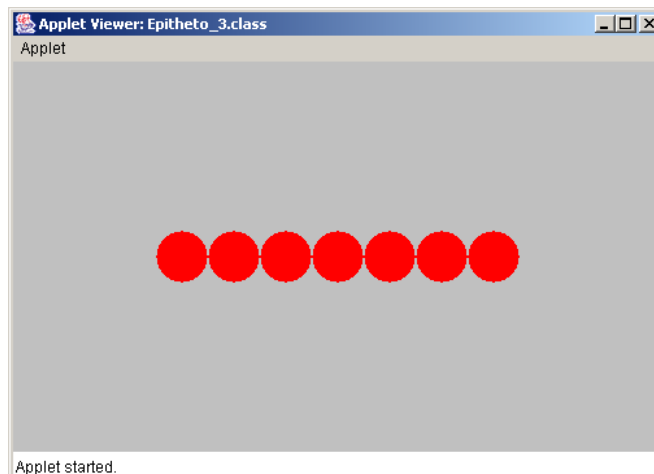
Δημιουργήστε ένα applet που θα χρωματίζει το background με χρώμα κυανό. Θα παίρνει τις συντεταγμένες τις Οθόνης από το σύστημα και θα δημιουργεί ένα ορθογώνιο χρώματος μπλέ γύρο από την οθόνη όπως φαίνεται και στο σχήμα. Στη συνέχεια θα δημιουργεί την πρόταση "I'm a simple applet" σε χρώμα γραμματοσειράς κόκκινο.

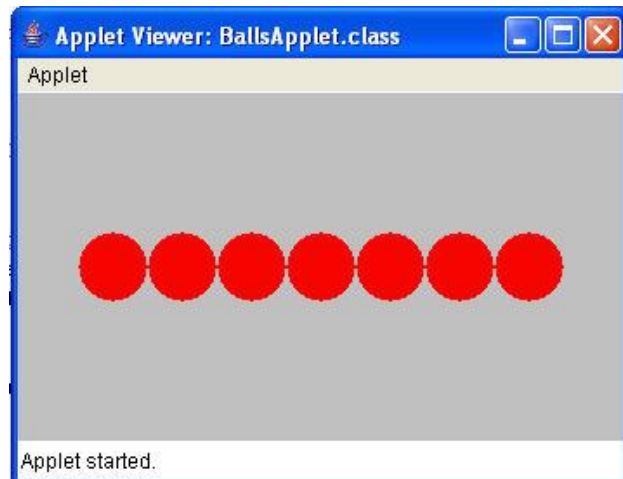


```
import java.applet.Applet;
import java.awt.Graphics;
import java.awt.Color;
public class SimpleApplet extends Applet
{
    String text = "I'm a simple applet";
    public void init() {
        text = "I'm a simple applet";
        setBackground(Color.cyan);
    }
    public void start() {
        System.out.println("starting...");
    }
    public void stop() {
        System.out.println("stopping...");
    }
    public void destroy() {
        System.out.println("preparing to unload...");
    }
    public void paint(Graphics g) {
        System.out.println("Paint");
        g.setColor(Color.blue);
        g.drawRect(0, 0,
            getSize().width -1,
            getSize().height -1);
        g.setColor(Color.red);
        g.drawString(text, 15, 25);
    }
}
```

## Άσκηση 6η

Δημιουργήστε το project applet με αρχείο Balls. Σκοπός της άσκησης είναι να τροποποιήσετε το applet ώστε να εκτυπώνει μία γραμμή από μπάλες δεξιά της πρώτης που σας δίνεται. Το συνολικό πλήθος από μπάλες το δίνει ο χρήστης, εδώ έδωσε 7





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

public class Game extends Applet {
    private int diameter = 40;    //diameter of balls
    private int[] x;              //x-coordinate of balls
    private int[] y;              //y-coordinate of balls
    private int n;                //number of balls
    private int width, height;    //applet's width and height

    public void init() {
        //find applet's width and height
        width = Integer.parseInt(getParameter("width"));
        height = Integer.parseInt(getParameter("height"));

        //interact with user
        String in = JOptionPane.showInputDialog(null, "Give a
series of balls: ");
        in = in.trim();
        n = Integer.parseInt(in);
        if(n > 0){
            x = new int[n];        //initialize array
            y = new int[n];        //initialize array
        }
    }

    public void paint(Graphics g) {
        //set background color
        setBackground(Color.lightGray);

        //set color
        g.setColor(Color.red);

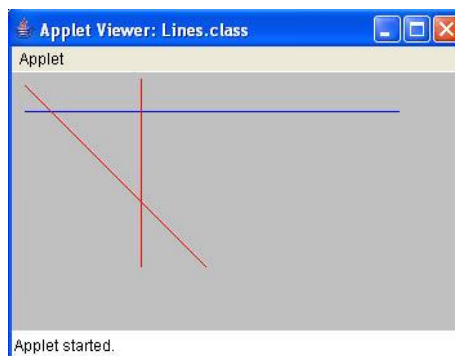
        //draw balls
        if(n > 0){
            int length = n*diameter;    //total length
of balls
            x[0] = (width - length)/2;    //find x-
coordinate of first ball
```

```
        y[0] = (height - diameter)/2; //find y-coordinate  
of first ball  
        g.fillOval(x[0], y[0], diameter, diameter);  
        for(int i=1; i<n; i++){  
            x[i] = x[i-1] + diameter;  
            y[i] = y[0];  
            g.fillOval(x[i], y[i], diameter, diameter);  
        }  
    }  
}
```

## Άσκηση 7η

Να δημιουργήσετε ένα applet με όνομα Lines που θα σχεδιάζει μία οριζόντια γραμμή χρώματος μπλέ, μια κατακόρυφη και μία γραμμή υπό γωνία χρώματος κόκκινου όπως φαίνεται στο σχήμα.

Σκοπός μας να κάνουμε χρήση της μεθόδου: **drawLine(int x1, int y1, int x2, int y2)**.



```
import java.awt.*;  
import java.applet.*;  
import javax.swing.JOptionPane;  
  
public class Lines extends Applet {  
    //specify 3 rows with 2 points each  
    private int[][] coords = {{10, 30, 300, 30}, {100, 5, 100, 150},  
        {10, 10, 150, 150}};  
  
    public void init() {  
        setBackground(Color.lightGray);  
    }  
  
    public void paint(Graphics g) {  
        //set color  
        g.setColor(Color.blue);  
  
        //draw horizontal line
```



```
g.drawLine(coords[0][0], coords[0][1], coords[0][2],  
coords[0][3]);  
  
//set color  
g.setColor(Color.red);  
//draw vertical line  
g.drawLine(coords[1][0], coords[1][1], coords[1][2],  
coords[1][3]);  
//g.setColor(Color.black);  
g.drawLine(coords[2][0], coords[2][1], coords[2][2],  
coords[2][3]);  
}  
}
```

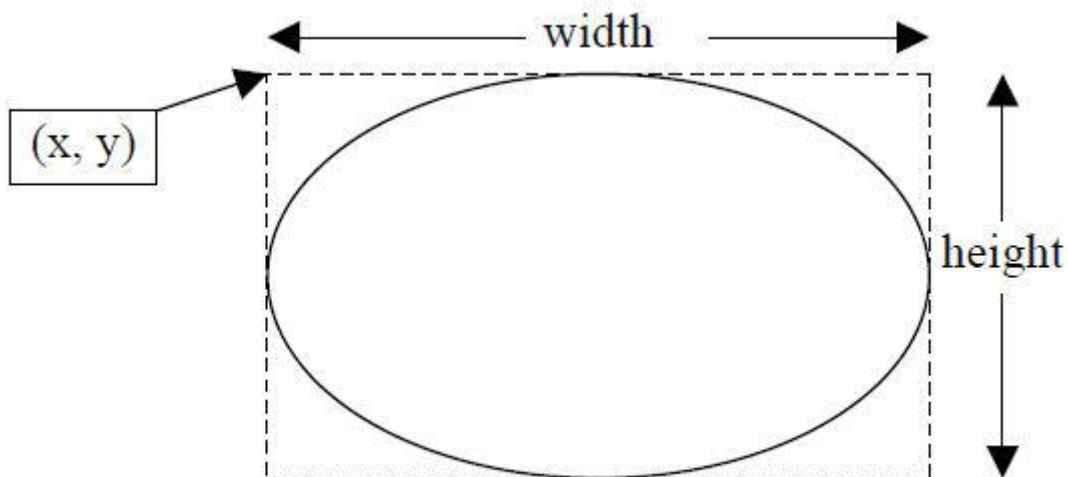
## Άσκηση 8η

Να δημιουργήσετε ένα applet με όνομα DrawShapes που θα σχεδιάζει μία έλλειψη, ένα κύκλο γεμάτο με χρώμα, ένα τρίγωνο και ένα ορθογώνιο γεμάτο με χρώμα όπως φαίνεται στο σχήμα.

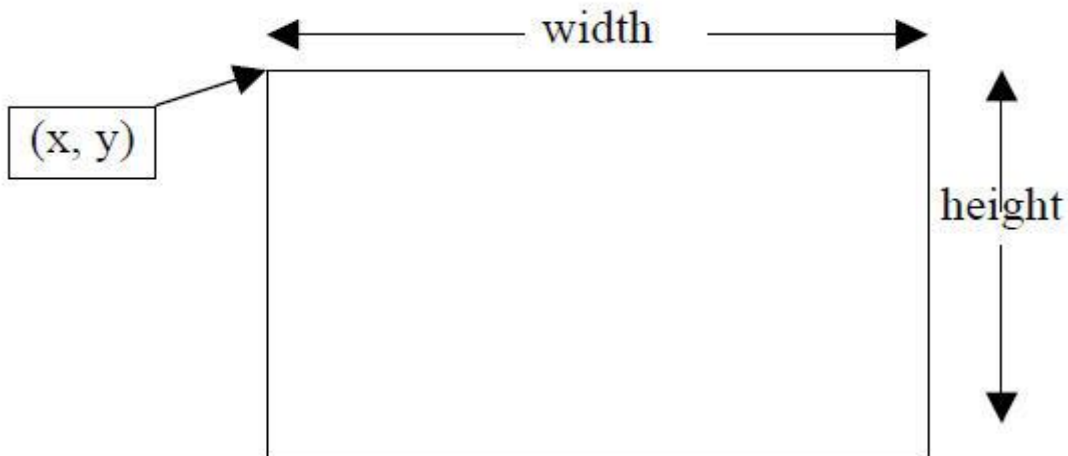
Σκοπός μας να κάνουμε χρήση των μεθόδων:

- `drawOval(int x, int y, int width, int height)`
- `fillOval(int x, int y, int width, int height)` //γεμίζει το Oval
- `drawPolyline(int[] xPoints, int[] yPoints, int nPoints)`
- `fillRect(int x, int y, int width, int height)`

Η `drawOval(int x, int y, int width, int height)` δέχεται τις συντεταγμένες  $(x, y)$  της πάνω αριστερής γωνίας του περιγεγραμμένου ορθογωνίου της έλλειψης και το πλάτος και το ύψος της είναι :



Η **fillRect(int x, int y, int width, int height)** δέχεται τις συντεταγμένες (x, y) της πάνω αριστερής γωνίας του ορθογωνίου της έλλειψης και το πλάτος και το ύψος της είναι :



```
import java.awt.*;
import java.applet.*;
import javax.swing.JOptionPane;
public class DrawShapes extends Applet {
    public void init() {
        setBackground(Color.lightGray);
    }

    public void paint(Graphics g) {
        //set color
        g.setColor(Color.blue);

        //draw ellipsis
        g.drawOval(50, 50, 180, 100);

        //set color
        g.setColor(Color.red);

        //fill circle
        g.fillOval(200, 50, 100, 100);
        //set color
        g.setColor(Color.black);
        //draw triangle
        int x[] = {40, 200, 320, 40};
        int y[] = {180, 40, 160, 180};
        g.drawPolyline(x, y, x.length);

        //set color
        g.setColor(Color.magenta);

        //fill rectangle
        g.fillRect(30, 30, 80, 50);
    }
}
```

## Άσκηση 9η

Να δημιουργήσετε ένα applet με όνομα Hello που ζητά το όνομά σας και μετά το εκτυπώνει



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

public class HiString extends Applet
{
    private String str;
    static boolean iterations = false;

    public void init() {
        setBackground(Color.lightGray);
        //interact with user
        String name = JOptionPane.showInputDialog(null, "Give me your
name: ");
        str = new String("Welcome " + name + ". This is a Java
Applet!!");
    }

    public void paint(Graphics g) {
        //set color
        g.setColor(Color.red);
        //set font
        Font f = new Font("Serif", Font.BOLD+Font.ITALIC, 20);
        g.setFont(f);
    }
}
```

```
//examine if an event triggers method paint() repeatedly
if (!iterations)
    JOptionPane.showMessageDialog(null, "Font = " +
f.toString());
iterations = true;

//write string
g.drawString(str, 10, 50 );

}
}
```

## Γ. Ασκήσεις άλυτες για το εργαστήριο

### Άσκηση 4<sup>η</sup>

Να γράψετε ένα applet TestOval το οποίο να σχεδιάζει 10 ομόκεντρους κύκλους.  
(Δείτε στις Παρατηρήσεις τον ορισμό της μεθόδου drawOval())

### Άσκηση 5<sup>η</sup>

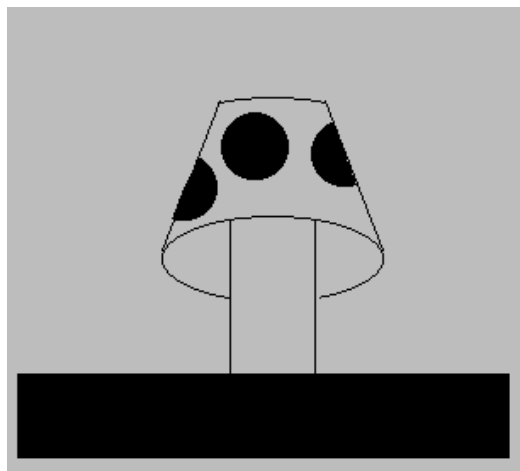
Να γράψετε ένα applet TestRectangle το οποίο να ρωτά τον χρήστη πόσα ορθογώνια να σχεδιάσει και στη συνέχεια να σχεδιάζει τόσα τυχαία ορθογώνια με τυχαία χρώματα.

### Άσκηση 6<sup>η</sup>

Να γράψετε ένα applet TestPolygon2 το οποίο να σχεδιάζει ένα κανονικό εξαγώνο.

### Άσκηση 7<sup>η</sup>

Να γράψετε ένα applet DrawLamp το οποίο να σχεδιάζει την παρακάτω λάμπα.



## Παρατηρήσεις Applet

Τα applet είναι κώδικας γραμμένος σε Java και δεν μπορούν να τρέξουν άμεσα αλλά μόνο αν κληθούν από ένα HTML αρχείο σε κάποιο browser (Netscape, Internet explorer κλπ) ή τον appletviewer.

Πως φτιάχνουμε ένα applet

- Δημιουργία java κώδικα. Η κλάση που θα περιέχει τον ορισμό του νέου applet θα κληρονομεί την κλάση Applet που είναι ήδη ορισμένη στην Java με κάποια βασικά χαρακτηριστικά και κάποιες μεθόδους όπως θα δείτε παρακάτω
- Δημιουργία html αρχείου
- Εκτέλεση από explorer ή appletviewer

Όταν δημιουργώ μέσα από το JCreator νέο applet ταυτόχρονα δημιουργείτε και το Html αρχείο που περιέχει το applet.

Ο ορισμός της κλάσης ενός applet έχει ως εξής

```
public class HelloWorldApplet extends Applet
```

Η λέξη κλειδί extends υπονοεί ότι αυτή η κλάση είναι μια υποκλάση της κλάσης Applet. Ή αλλιώς η Applet είναι η υπερκλάση του HelloWorldApplet. Η κλάση Applet προσδιορίζεται στο πακέτο java.applet.Applet. Εφόσον το HelloWorldApplet είναι υποκλάση του Applet, τότε το HelloWorldApplet κληρονομεί όλη τη λειτουργικότητα του αρχικού Applet. Ό,τι μπορεί να κάνει η κλάση Applet, μπορεί να το κάνει και το HelloWorldApplet επίσης.

### Μέθοδοι που χρησιμοποιεί ένα applet

Για την δημιουργία Java εφαρμογών, η κλάση της εφαρμογής πρέπει να διαθέτει την μέθοδο main(). Όταν η εφαρμογή αρχίζει να τρέχει, εκτελείται η μέθοδος main() η οποία καθορίζει την συμπεριφορά του προγράμματος. Αντιθέτως στα Applets προγράμματα υπάρχουν διαφορετικές λειτουργίες που αντιστοιχούν σε γεγονότα που συμβαίνουν κατά τη διάρκεια της ζωής του Applet (πχ. γεγονότα αρχικοποίησης, ζωγραφικής, ποντικιού κλπ). Σε κάθε λειτουργία αντιστοιχεί και κάποια μέθοδος η οποία καλείται από τον Browser όταν ένα γεγονός συμβεί. Η μέθοδοι των λειτουργιών έτσι όπως ορίζονται στην Applet κλάση της Java δεν κάνουν τίποτε. Για να δώσουμε κάποια συμπεριφορά σε κάποιο γεγονός της Applet εφαρμογή μας πρέπει να ξαναορίσουμε την μέθοδο που αντιστοιχεί στο γεγονός (τεχνική method overriding) μέσα στην υποκλάση της Applet που δημιουργήσαμε. Οι πέντε πιο βασικές μέθοδοι μιας applet είναι:

- public void init() : κατά την φόρτωση του applet, αρχικές τιμές, πέρασμα παραμέτρων, φόρτωση εικόνων & γραμματοσειρών
- public void start() : μετά την init(), ή σε αλλαγή σελίδας και επιστροφή, μπορεί να εκτελεσθεί πολλές φορές
- public void stop() : όταν θέλουμε να σταματήσουμε κάποιες εργασίες που ξεκίνησαν με την start() ή όταν ο χρήστης αλλάζει σελίδα
- public void destroy(): καλείται από την stop() όταν κλείνει το πρόγραμμα ανάγνωσης ιστοσελίδων και γίνεται απελευθέρωση των πόρων του συστήματος που είχαν δεσμευτεί από το applet
- public void paint(Graphics g): καλείται κάθε φορά που χρειάζεται να σχεδιαστεί κάτι στην οθόνη, πχ μερική επικάλυψη παραθύρου από άλλο. Η μέθοδος paint χρησιμοποιεί ένα αντικείμενο Graphics που

επιλέγουμε να το λέμε `g`. Η κλάση `Graphics` ορίζεται στο πακέτο `java.awt.Graphics`

## Γραφικά

Το πακέτο `Abstract Window Toolkit AWT`, και το πακέτο `swing` που είναι νεώτερο του `AWT` και πιο πλούσιο. Με την εντολή `import java.awt.*;` ή `import javax.swing.*;` κάνω `import` όλων των κλάσεων του πακέτου για να είναι διαθέσιμες στο `applet` που έχω δημιουργήσει.

## Χρώμα

Στο πακέτο `awt` υπάρχει η τάξη `Color` με την οποία ρυθμίζουμε τα χρώματα στην οθόνη. (RGB- Red,Green,Blue 0-255). Υπάρχουν και τα έτοιμα χρώματα π.χ. `blue`, `red`, `white` κ.λ.π.

Μέθοδοι για ορισμό χρώματος στο

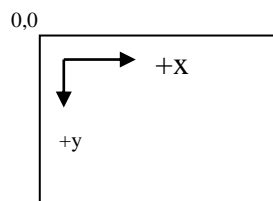
Προσκήνιο : `setForeground(Color.gray)`

Παρασκήνιο : `setBackground(Color.cyan)`

Χρήση : `g.setColor(Color.yellow)`

## Σύστημα συντεταγμένων

Η σχεδίαση ενός αντικειμένου στην οθόνη, γίνεται μέσω μεθόδων της κλάσης `Graphics`. Όλες οι μέθοδοι σχεδίασης παίρνουν ορίσματα που αφορούν σημεία, γωνίες ή αρχικά σημεία των αντικειμένων στο σύστημα συντεταγμένων μιας `applet`. Το σύστημα συντεταγμένων έχει την αρχή του (0,0) στην αριστερή άνω γωνία. Οι θετικές τιμές του `x` βρίσκονται στα δεξιά και οι θετικές `y` τιμές είναι προς τα κάτω.



## Γεωμετρικά σχήματα

Μέθοδοι για να ζωγραφίσουμε γεωμετρικά σχήματα

`g.drawLine(x1,y1,x2,y2)` -> ζωγραφίζει ένα ευθύγραμμο τμήμα από το σημείο `x1,y1` ως το σημείο `x2,y2`

`g.drawRect(x1,y1,πλάτος, ύψος)` -> Παραλληλόγραμμα

`g.drawOval(x1,y1,πλάτος, ύψος)` -> έλλειψη

`g.drawArc(x1,y1,πλάτος, ύψος, αρχική γωνία, άνοιγμα γωνίας)` -> Τόξο

`g.drawPolygon(int xpoints[],int ypoints[],int points)` -> Πολύγωνο

## Βιβλιογραφία

- Γιώργος Λιακέας, "Εισαγωγή στη JAVA 2", Κλειδάριθμος, 2003.
- Deitel H. M., Deitel P. J., "JAVA: How To Program", Pearson Education Inc., USA, 2003 (Fifth edition).