

ΠΡΟΤΕΙΝΟΜΕΝΕΣ
ΛΥΣΕΙΣ ΑΣΚΗΣΕΩΝ
Πολυμορφισμός –
Αφηρημένες Κλάσεις (*Polymorphism –
Abstract Classes*)

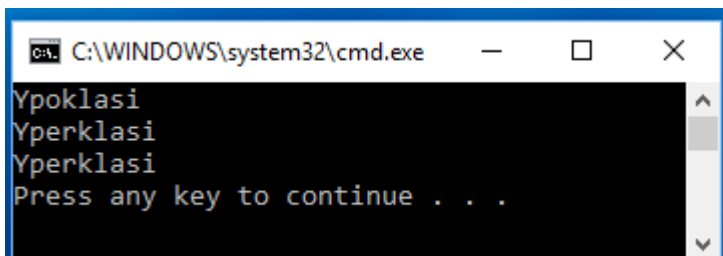
ΑΣΚΗΣΗ-1^η

Απόκρυψη Πεδίων (*Hiding Fields*)

```
class SuperClass
{ String f = "Υperklasi";}

class SubClass extends SuperClass
{String f = "Υpoklasi";}

class ΥpervasiPediou {
    public static void main(String[] args) {
        SubClass p1 = new SubClass();
        System.out.println(p1.f);
        SuperClass p2 = new SubClass(); //αναφορά υπερκλάσης σε αντικ. υποκλάσης
        System.out.println(p2.f); //πρόσβαση
        System.out.println(((SuperClass)p1).f); //casting
    }
}
```



```
C:\WINDOWS\system32\cmd.exe
Υpoklasi
Υperklasi
Υperklasi
Press any key to continue . . .
```

ΑΣΚΗΣΗ-2^η

Στατική και Δυναμική δέσμευση

```
class Employee{
    public void Salary(){
        System.out.println("O Employee plironetai"); } }
class Misthotos extends Employee{
    public void Salary(){
        System.out.println("O misthotos plironetai");
    }
}

class Test {
    public static void main( String args[]){
        Employee obj1 = new Misthotos();
        obj1.Salary(); //static binding

        Employee obj2 = new Employee();
        obj2.Salary(); //dynamic binding
    }
}
```

A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system...'. The command prompt displays the output of the Java program: 'O misthotos plironetai' on the first line, 'O Employee plironetai' on the second line, and 'Press any key to continue . . .' on the third line. The cursor is positioned at the end of the third line.

ΑΣΚΗΣΗ-3^η (Τυπικό Παράδειγμα Πολυμορφισμού)

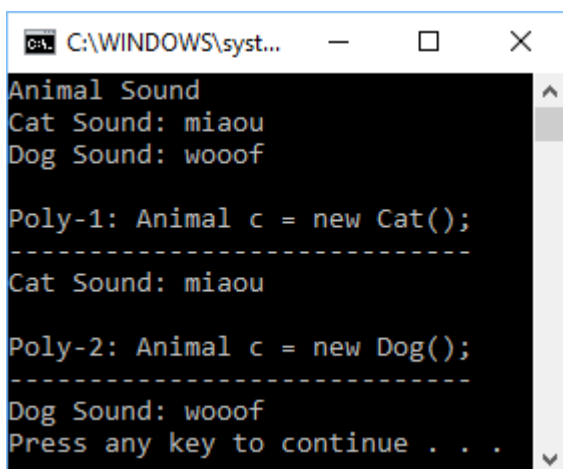
```
class Animal {
    void sound() {System.out.println("Animal Sound");}
}
class Cat extends Animal {
    void sound() { // Overriding
        System.out.println("Cat Sound: miaou"); }
}
```

```

class Dog extends Animal {
    void sound() {// Overriding
        System.out.println("Dog Sound: woof");} }

class testPoly {
    public static void main(String[] args) {
        Animal animal = new Animal();
        animal.sound();
        Cat cat = new Cat();
        cat.sound();
        Dog dog = new Dog();
        dog.sound();
        System.out.println();
        System.out.println("Poly-1: Animal c = new Cat();");
        System.out.println("-----");
        Animal c = new Cat(); //ότι η κλάση και όχι η αναφορά
        c.sound(); // η cat sound()
        System.out.println();
        System.out.println("Poly-2: Animal c = new Dog();");
        System.out.println("-----");
        Animal d = new Dog(); //ότι η κλάση και όχι η αναφορά
        d.sound(); //dog sound()
    } }

```



```

C:\WINDOWS\system...
Animal Sound
Cat Sound: miaou
Dog Sound: woof

Poly-1: Animal c = new Cat();
-----
Cat Sound: miaou

Poly-2: Animal c = new Dog();
-----
Dog Sound: woof
Press any key to continue . . .

```

AΣΚΗΣΗ-4ⁿ (Upcasting - Downcasting)

```
class Camera {
    public void start(){
        System.out.println("Camera started");} }

class Nikon extends Camera {
    public void start(){
        System.out.println("Nikon started"); }
    public void snap() {
        System.out.println("Photo taken"); }}

class TestUpcastingDowncasting {
    public static void main( String args[]){
        Camera c1=new Camera();
        Nikon n1=new Nikon();
        c1.start();
        n1.start();
        n1.snap();

        //upcasting
        System.out.println("-----");
        System.out.println("Upcasting");
        System.out.println("-----");
        Camera c2=new Nikon();
        c2.start();

        //downcasting
        System.out.println("-----");
        System.out.println("Downcasting");
        System.out.println("-----");

        Camera c3=new Nikon();
        Nikon n2 = (Nikon)c3;
        n2.start();
        n2.snap();
    }
}
```

```
C:\Windows\sy...
Camera started
Nikon started
Photo taken
-----
Upcasting
-----
Nikon started
-----
Downcasting
-----
Nikon started
Photo taken
Press any key to continue . .
```

ΑΣΚΗΣΗ-5^η (Υπέρβαση μεθόδων- RunTime Polymorphism)

```
class Human{
    public void eat() {System.out.println(" O anthropos troei pizza");}
    public void drink() {System.out.println(" O anthropos pinei byra");}}

class Boy1 extends Human{
    public void eat(){System.out.println(" To agori troei pizza");}
    public void drink(){System.out.println(" To agori pinei byra");}

    public void A_method() {
        Boy1 b = new Boy1();
        Human h=new Boy1();
        System.out.println("To antikeimeno (boy) b.eat():");
        b.eat();
        System.out.println("O Human (super).drink():");
        super.drink();
        System.out.println("To antikeimeno(Human) h.eat():");
        h.eat(); //υπέρβαση-overriding, Runtime polymorphism
    }

    public static void main( String args[] ) {
        Boy1 obj = new Boy1();
        obj.A_method(); }}
```

```
C:\WINDOWS\system... - □ ×
To antikeimeno (boy) b.eat():
  To agori troei pizza
O Human (super).drink():
  O anthros pinei byra
To antikeimeno (Human) h.eat():
  To agori troei pizza
Press any key to continue . . .
```

ΑΣΚΗΣΗ-6^η (Υπέρβαση μεθόδων- Πολυμορφισμός Υποκατάστασης)

```
class Circle {
    private double radius;

    public Circle(){};

    public Circle(double radius) {this.radius = radius;}

    public double getRadius() {return this.radius;}

    public double getArea(){return radius * radius * Math.PI;}

    public String toString() {
        return "Circle[radius=" + radius + "]; } }

class Cylinder extends Circle {
    private double height;

    public Cylinder() {
        super();
        height = 1.0; }

    public Cylinder(double height) {
        super();
        this.height = height;}

    public Cylinder(double radius, double height) {
        super(radius);
        this.height = height;}

    public double getHeight() {return height;}

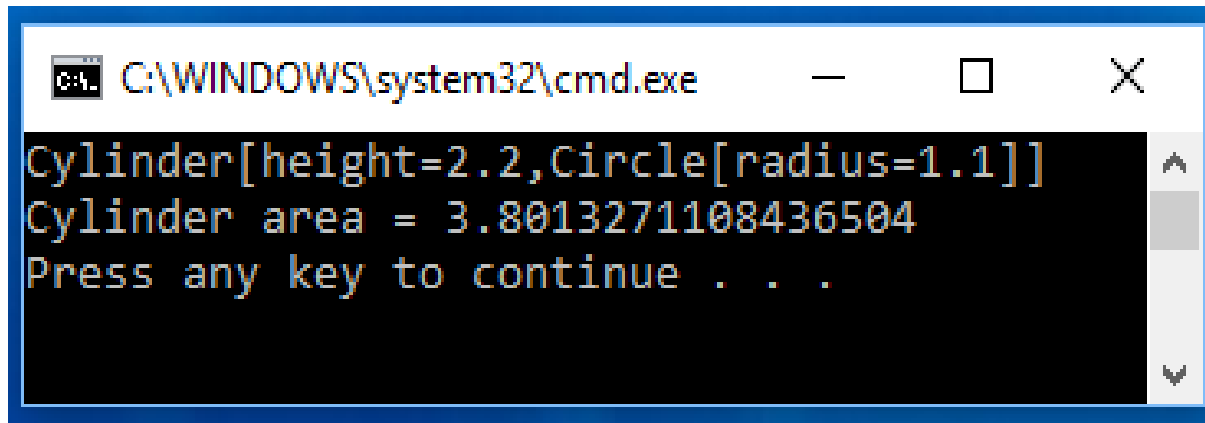
    public double getVolume() {return getArea()*height;}

    public String toString() {
        return "Cylinder[height=" + height + "," + super.toString() + "];
    } }
} }
```

```

class TestCylinder {
    public static void main (String[] args) {
        // Υποκατάσταση αντικειμένου της υπερκλάσης με αντικείμενο της υποκλάσης
        Circle c1 = new Cylinder(1.1, 2.2);
        System.out.println(c1.toString()); // overridden
        System.out.println("Cylinder area = "+c1.getArea()); // overridden
    }
}

```



```

C:\WINDOWS\system32\cmd.exe
Cylinder[height=2.2,Circle[radius=1.1]]
Cylinder area = 3.8013271108436504
Press any key to continue . . .

```

ΑΣΚΗΣΗ-7^η (κληρονομικότητα με αφηρημένη υπερκλάση)

```

abstract class Shape {
    private String color;
    private boolean filled;
    public abstract double getArea();
    public abstract double getPerimeter(); }

class Circle extends Shape {
    private double radius;
    public Circle(double r) {this.radius = r;}
    public void setRadius(double r) {this.radius = r;}
    public double getRadius() {return radius;}
    @Override /** Return area */
    public double getArea()
        {return radius * radius * Math.PI;}
    public double getDiameter()
        {return 2 * radius;}
    @Override /** Return perimeter */
    public double getPerimeter()
        {return 2 * radius * Math.PI;}
}

```

```
public void printCircle() {  
    System.out.println("The radius is " + radius);  
};}
```

```
class Rectangle extends Shape {  
    private double width;  
    private double height;  
public Rectangle(double width, double height) {  
    this.width = width;  
    this.height = height;  
public double getWidth()  
    {return width;}  
public void setWidth(double width)  
    {this.width = width;}  
public double getHeight()  
    {return height;}  
public void setHeight(double height)  
    {this.height = height;}  
@Override /** Return area */  
public double getArea()  
    {return width * height;}  
@Override /** Return perimeter */  
public double getPerimeter()  
    {return 2 * (width + height);}}
```

```
class TestShape {  
    public static void main(String[] args) {  
        Shape Object1 = new Circle(8);  
        Shape Object2 = new Rectangle(2, 4);  
// Display circle  
        displayObject(Object1);  
// Display rectangle  
        displayObject(Object2); }  
public static void displayObject(  
    Shape object) {  
        System.out.println("The area is " + object.getArea());  
        System.out.println("The perimeter is " + object.getPerimeter()); } }
```



```
C:\WINDOWS\system32\...
The area is 201.06192982974676
The perimeter is 50.26548245743669
The area is 8.0
The perimeter is 12.0
Press any key to continue . . .
```

ΑΣΚΗΣΗ-8^η (κληρονομικότητα με αφηρημένη υπερκλάση)

```
abstract class Employee {
    protected String name;
    protected int bonus; //bonus
    protected int hours; //hours
    protected int payType; //0=salary, 1=byhour
    Employee(String s, int b, int h, int p)
        {name=s; bonus=b; hours=h; payType=p;}
    public String getName() {return name;}
    public String getPayType() {
        String pType;
        if (payType==0) pType="Misthos";
        else Type="Oromisthios";
        return pType; }

    abstract String getEmpType();
    abstract void calcSalary();
}

class Administrator extends Employee {
    Administrator(String s, int b, int h, int p)
        {super(s,b,h,p);}
    public String getEmpType()
        {return "Dioikitikos Ypalilos";}
    public void calcSalary() {
        int s=0;
        if (payType==0) s=1200+bonus;
        //vasikos=1200 + bonus
        else s=(hours*12); //12 Euro per hour
        System.out.println(" Misthos Dioikitikou= " + s); }}
```

```

class Technical extends Employee {
    public Technical(String s, int b, int h, int p)
    {super(s,b,h,p);}
    public String getEmpType()
    {return "Tehnikos Ypalilos";}
    public void calcSalary() {
int s=0;
if (payType==0) s=800+bonus;
    //vasikos=800 + bonus
else s=(hours*10); //10 Euro per hour
    System.out.println(" Misthos Tehnikou = "+s);
    } }
class Company {
    Employee emp[]= new Employee[4];
    public void setEmployee(Employee e,int a)
    {emp[a]=e;}
    public void printAll() {
        for (int i=0;i<4;i++) {
            System.out.println();
            System.out.println("Onoma      = "+ emp[i].getName());
            System.out.println("Typos Ypalilou = "+ emp[i].getEmpType());
            System.out.println("Typos Pliromis = "+
                emp[i].getPayType());
            emp[i].calcSalary(); } } }
class TestEmployeeAbstract {
    public static void main(String[] args) {
        int a;
        Employee e1=new Technical("Nikas",1000,10,0);
        Employee e2=new Administrator("Vasileiou",1000,20,0);
        Employee e3=new Technical("Paylidis",10,10,1);
        Employee e4=new Administrator("Apostolatos",10,10,1);
        Company c=new Company();
        c.setEmployee(e1,0);
        c.setEmployee(e2,1);
        c.setEmployee(e3,2);
        c.setEmployee(e4,3);
        c.printAll(); } }

```

```
C:\WINDOWS\system32\cm... - □ ×
Onoma      = Nikas
Typos Ypalilou = Tehnikos Ypalilos
Typos Pliromis = Misthos
Misthos Tehnikou = 1800

Onoma      = Vasileiou
Typos Ypalilou = Dioikitikos Ypalilos
Typos Pliromis = Misthos
Misthos Dioikitikou= 2200

Onoma      = Paylidis
Typos Ypalilou = Tehnikos Ypalilos
Typos Pliromis = Oromisthios
Misthos Tehnikou = 100

Onoma      = Apostolatos
Typos Ypalilou = Dioikitikos Ypalilos
Typos Pliromis = Oromisthios
Misthos Dioikitikou= 120
Press any key to continue . . .
```

ΑΣΚΗΣΗ-9^η (κληρονομικότητα με αφηρημένη υπερκλάση)

```
abstract class MousikaOrgana {
    int i;
    public abstract void play();
    public String BrandName() {
        return "Mousiko Organo ";;}}

class Kithara extends MousikaOrgana {
    public void play() {
        System.out.println ("Paizei Kithara"); }
    public String BrandName() {
        return "Kithara";;}}

class HlektrikiKithara extends Kithara {
    public void play(){
        System.out.println("Paizei Hlektriki Kithara"); }
    public String BrandName() {
        return "Marka: Gibson";;}}

class KlassikiKithara extends Kithara {
    public void play(){
        System.out.println("Paizei Klassiki Kithara");
    }
}
```

```

public String BrandName() {
    return "Marka: Ovation";}}
class Krousta extends MousikaOrgana {
    public void play(){
        System.out.println("Paizei Krousta"); }
    public String BrandName(){
        return "Marka: Ludwing";}}
public class MusicOrgans{
    static void tune(MousikaOrgana mo) {
        mo.play() ;}

public static void main(String[] args) {
    MousikaOrgana[] o = new MousikaOrgana[4];
    int i = 0;
    o[i++] = new Kithara();
    o[i++] = new HlektrikiKithara();
    o[i++] = new KlassikiKithara();
    o[i++] = new Krousta() ;
    for(i = 0; i < o.length; i++) {
        tune(o[i]);
        System.out.println( o[i].BrandName() );
    } } }

```