



Αντικειμενοστρεφής Προγραμματισμός

Ενότητα 2: Κλάσεις, Αντικείμενα και η Σύνταξη της
Java

Γρηγόρης Τσουμάκας, Επικ. Καθηγητής
Τμήμα Πληροφορικής



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

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 παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο την αναδιαμόρφωση του εκπαιδευτικού υλικού.
- 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.





ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΙΚΑ
ΜΑΘΗΜΑΤΑ



Κλάσεις, Αντικείμενα και η Σύνταξη της Java



Τα παραδείγματα κώδικα που χρησιμοποιούνται σε κάποιες από τις ακόλουθες διαφάνειες μπορούν να βρεθούν στον παρακάτω σύνδεσμο:
<http://users.auth.gr/greg/oop.zip>

Σύνταξη Πηγαίου Κώδικα

- Αγνοούνται τα λευκά κενά (white space).
 - Μορφοποίηση κώδικα, σύμφωνα με τις γνωστές από τη C/C++ και διαδεδομένες συμβάσεις.
- Ευαισθησία πτώσης (case sensitivity).
 - Π.χ. **FOR** και **For** διαφέρουν από την εντολή **for**
 - Π.χ. μεταβλητή **counter** διαφέρει από την **counteR**
- Ο πηγαίος κώδικας Java γράφεται σε Unicode.
 - Χρήση χαρακτήρων Unicode στα σχόλια, στα ονόματα μεταβλητών, κ.α.



Παράδειγμα Πηγαίου Κώδικα

```
public class Πρόγραμμα {  
  
    public static void main(String[] args) {  
        // δήλωση μιας μεταβλητής  
        int μεταβλητή = 5;  
        int αποτέλεσμα = μεταβλητή + 3;  
        System.out.println(αποτέλεσμα);  
        System.out.println("Πρόγραμμα");  
    }  
}
```

Πρόγραμμα.java



Σχόλια (1/2)

- Σχόλια μίας γραμμής.
 - Ξεκινούν από τη συμβολοσειρά `//` και μπορούν να φτάσουν μέχρι το τέλος της γραμμής.
 - `//` αυτό είναι ένα σχόλιο μίας γραμμής
- Σχόλια πολλών γραμμών.
 - Μεταξύ των χαρακτήρων `/*` και `*/`
 - Δεν επιτρέπεται να είναι φωλιασμένα.
 - `/*` σχόλιο πολλών γραμμών `*/`

Comments.java



Σχόλια (2/2)

- Σχόλια τεκμηρίωσης (εργαλείο **javadoc**).
 - Μεταξύ των χαρακτήρων `/**` και `*/`
 - Παράγονται ιστοσελίδες με τεκμηρίωση.
 - `/**`
 - * `@param n an integer value`
 - * `@return the square value of the input`
 - *
 - * `@author Grigorios Tsoumakas`
 - * `/`
 - Θα τα εξετάσουμε αναλυτικά αργότερα...



Σύνταξη Αναγνωριστικών

- Αναγνωριστικά (identifiers).
 - Ονόματα μεταβλητών, κλάσεων, μεθόδων κτλ.
- Κανόνες σύνταξης αναγνωριστικών.
 - Απεριόριστη ακολουθία από γράμματα και ψηφία του συνόλου Unicode.
 - Δεν πρέπει να ξεκινάει με ψηφίο.
 - Δεν επιτρέπονται χαρακτήρες στίξης πέραν του '_', του '\$' και άλλων συμβόλων νομισμάτων.
 - Δεν επιτρέπονται δεσμευμένες λέξεις.



Συμβάσεις για τα Αναγνωριστικά

- Ονόματα κλάσεων ξεκινούν με κεφαλαίο.
 - Τα επόμενα γράμματα είναι μικρά, π.χ. `Student`
- Ονόματα μεταβλητών/μεθόδων με μικρό.
 - Τα επόμενα γράμματα είναι μικρά, π.χ. `age`, `insert`
- Αν αποτελούνται από παραπάνω λέξεις.
 - Το πρώτο γράμμα κάθε επόμενης λέξης γίνεται κεφαλαίο, π.χ. `MyClass`, `dateOfBirth`
- Για σταθερές μεταβλητές.
 - Χρησιμοποιούμε κεφαλαία, και χωρίζουμε τις λέξεις με `'_'`, π.χ. `NUMBER_OF_EXPERIMENTS`



Ορισμός Κλάσης στην Java

```
public class Όνομα Κλάσης.
```

```
{
```

```
    // Πεδία (Fields)
```

```
    // Κατασκευαστές (Constructors)
```

```
    // Μέθοδοι (Methods)
```

```
}
```

μέλη της
κλάσης
(class
members)

Αποθηκεύουμε τον παραπάνω κώδικα στο
αρχείο κειμένου «ΌνομαΚλάσης.java»



Παράδειγμα Κλάσης 1

- Αυτόματος πωλητής εισιτηρίων (απλός).
 - Έκδοση ενός μόνο είδους εισιτηρίου με κάποια συγκεκριμένη τιμή.
 - Εισαγωγή χρημάτων.
 - Εκτύπωση εισιτηρίου.
 - Καταγραφή του συνολικού ποσού των χρημάτων που έχει συλλέξει.



Πεδία

- Χώροι μνήμης για την αποθήκευση των δεδομένων κάθε αντικειμένου.
- Καλούνται και **μεταβλητές στιγμιοτύπου**.
- Το σύνολο των τιμών όλων των πεδίων ενός αντικειμένου καλείται και **κατάσταση** του αντικειμένου.
- Μπροστά από τον τύπο δεδομένων τους βάζουμε (σχεδόν) πάντα τη λέξη κλειδί **private**.
 - Θυμηθείτε την αρχή της ενθυλάκωσης.



Τύποι Δεδομένων

- Θεμελιώδεις τύποι (primitive types).
 - Αριθμοί, χαρακτήρες, λογικές τιμές.
- Τύποι αναφοράς (reference types).
 - Κλάσεις, πίνακες, απαριθμητοί τύποι, κ.α.



Θεμελιώδεις Τύποι (1/3)

Τύπος	Bits	Εύρος τιμών
byte	8	-128 ... 127
short	16	-32768 ... 32767
int	32	-2147483648 ... 2147483647
long	64	-9223372036854775808 ... 9223372036854775807
float	32	απλής ακρίβειας IEEE 754
double	64	διπλής ακρίβειας IEEE 754
char	16	'\u0000' ... '\uffff'
boolean	1	true, false



Θεμελιώδεις Τύποι (2/3)

- Boolean.
 - Ως τιμή παίρνει τις δεσμευμένες λέξεις `true`, `false`.
 - Δεν μετατρέπεται από/προς ακέραιο όπως στη C.
- Char.
 - Χαρακτήρες του συνόλου Unicode (16 bit).
 - `'ξ'`, `'b'`, `'\t'`, `'\u0000'`, `'\"'`, `'\''`, `'\\'`
- String (τύπος αναφοράς).
 - Κλάση για αποθήκευση συμβολοσειρών.
 - Π.χ. `"συμβολοσειρά"`, `"java"`, `"Sí señor"`.



Θεμελιώδεις Τύποι (3/3)

- Ακέραιοι.
 - Δεν υπάρχουν απρόσημοι όπως στις C/C++.
 - 1, -2, 100 (`int`).
 - 0xff, 0x- δεκαεξαδικό (`int`).
 - 0377, 0- οκταδικό (`int`).
 - 23L, -456l, 0xffL (`long`).
- Κινητής υποδιαστολής.
 - 0.0, 0.64, -34.56, 6.02E23 (`double`).
 - 0.123F, -4.05f, .01F, 1e-6f (`float`).



Κώδικας TicketMachinePlain (1/3)

```
/**  
 * Απλοϊκός αυτόματος πωλητής εισιτηρίων  
 */  
public class TicketMachinePlain  
{  
    // τιμή εισιτηρίου σε λεπτά του ευρώ  
    private int price;  
    // συνολικά λεπτά που έχουν εισαχθεί  
    private int total;  
}
```



Κατασκευαστές και Μέθοδοι

- Επικεφαλίδα.
 - Όνομα και προαιρετικά **παράμετροι**, μέσω των οποίων λαμβάνονται τιμές.
 - Η μεταβλητή στην οποία αποθηκεύεται η τιμή που λαμβάνεται καλείται **τυπική παράμετρος** ενώ η ίδια η τιμή καλείται **πραγματική παράμετρος**.
- Σώμα.
 - Αποτελείται από ένα ζεύγος αγκίστρων { και } με δηλώσεις και εντολές, το οποίο καλείται γενικότερα **μπλοκ** κώδικα.



Κατασκευαστές

- Εκτελούνται κατά τη δημιουργία αντικειμένων.
- Φροντίζουν για τη σωστή αρχική διαμόρφωση των αντικειμένων (**αρχικοποίηση**).
 - Θέτουν αρχικές τιμές στα πεδία, είτε σταθερές, είτε τιμές που δέχονται μέσω **παραμέτρων**.
- Έχουν **ίδιο όνομα με την κλάση** στην οποία ορίζονται.
- Μπροστά από το όνομα τους βάζουμε (σχεδόν) πάντα τη λέξη κλειδί **public**.



Ανάθεση Τιμών

- Με τον τελεστή =
 - `married = true;`
 - `age = 19;`
 - `x = y = z = 100;`
- Η έκφραση δεξιά από το = πρέπει να έχει τον ίδιο τύπο με την μεταβλητή αριστερά.
 - Εκτός από ορισμένες περιπτώσεις (προώθηση).



Κώδικας TicketMachinePlain (2/3)

```
public class TicketMachinePlain
{
    private int price;
    private int total;

    public TicketMachinePlain(int aPrice) {
        price = aPrice;
        total = 0;
    }
}
```



Προκαθορισμένος Κατασκευαστής

- Προκαθορισμένος κατασκευαστής.
 - Αν δεν καθορίσετε κατασκευαστή, τότε ο μεταγλωττιστής παρέχει αυτόματα έναν κενό κατασκευαστή χωρίς παραμέτρους.
- Παράδειγμα.
 - `public TicketMachinePlain() { }`
- Ποια θα είναι η κατάσταση του αντικειμένου;



Προκαθορισμένες Τιμές

- Όλα τα **πεδία** αρχικοποιούνται αυτόματα στις παρακάτω προκαθορισμένες τιμές.
 - Ωστόσο, αυτό είναι καλύτερα να αποφεύγεται, μέσω ρητής αρχικοποίησης σε κατασκευαστή.

Τύπος	Τιμή	Τύπος	Τιμή
byte	0	float	0.0f
short	0	double	0.0
int	0	char	'\u0000'
long	0L	boolean	false
αναφοράς	null		



Δημιουργία Αντικειμένου

- Παραδείγματα.
 - `MyClass a = new MyClass();`
 - `Student s = new Student("greg", 123);`
- Κλάσεις = τύποι αναφοράς.
 - Οι κλάσεις ορίζουν τύπους δεδομένων αναφοράς.
 - Τα αντικείμενα αποθηκεύονται σε μεταβλητές αυτών των τύπων δεδομένων.
- Δημιουργία αντικειμένου με τελεστή **new**.
 - Καλείται ο κατασκευαστής της κλάσης.



Τοπικές Μεταβλητές

- Προσωρινή αποθήκευση δεδομένων.
- Μπορούν να δηλώνονται σε οποιοδήποτε σημείο μιας μεθόδου ή ενός κατασκευαστή.
- Δηλώνονται όπως και τα πεδία, χωρίς όμως τη λέξη κλειδί `private` από μπροστά.
 - `int counter; boolean isRaining;` LocalVariables.java
- Υποχρεωτική ανάθεση τιμής πριν τη χρήση.
 - `int x, y; x = y + 1; // σφάλμα μεταγλώττισης`



Τυπικό Πρόγραμμα Java

- Τυπικό πρόγραμμα Java.
 - Ένα η παραπάνω αντικείμενα της ίδιας ή διαφορετικών κλάσεων που αλληλεπιδρούν.
- Θα πρέπει να υπάρχει μια εκτελέσιμη κλάση.
 - Κλάση που περιέχει μέθοδο με επικεφαλίδα `public static void main(String[] args)`.
 - Από εκεί ξεκινάει η εκτέλεση του προγράμματος.
 - Συνήθως εκεί δημιουργείται το πρώτο αντικείμενο, από το οποίο ξεκινάει η δημιουργία άλλων, κ.ο.κ.



Κώδικας TicketMachinePlainApp

```
public class TicketMachinePlainApp
{
    public static void main(String[] args) {
        TicketMachinePlain tmp1;
        tmp1 = new TicketMachinePlain(60);
    }
}
```



Είδη Μεταβλητών

- Μεταβλητές αντικειμένων (πεδία).
 - Αποθήκευση της κατάστασης του αντικειμένου.
 - Διαφορετική τιμή για κάθε αντικείμενο.
- Παράμετροι μεθόδων & κατασκευαστών.
 - Μεταβίβαση δεδομένων σε μεθόδους & κατασ/τές.
- Τοπικές μεταβλητές μεθόδων & κατασκευαστών.
 - Αποθήκευση προσωρινής κατάστασης.
- Μεταβλητές κλάσεων (στατικά πεδία).
 - Θα τις δούμε αναλυτικά παρακάτω.



Εμβέλεια Μεταβλητής (1/2)

- Εμβέλεια μεταβλητής.
 - Καθορίζει το τμήμα το πηγαίου κώδικα από όπου μπορεί να προσπελαστεί η μεταβλητή.
- Εμβέλεια ενός (private) πεδίου.
 - Όλος ο ορισμός της κλάσης στην οποία ανήκει.
 - Αν ήταν public, τότε θα ήταν προσβάσιμο και από το μπλοκ κώδικα όπου δηλώθηκε το αντικείμενο.



Εμβέλεια Μεταβλητής (2/2)

- Εμβέλεια παραμέτρου.
 - Το σώμα της μεθόδου ή του κατασκευαστή της.
- Εμβέλεια τοπικής μεταβλητής.
 - Από το σημείο δήλωσης της μέχρι τη λήξη του block στο οποίο δηλώθηκε.
 - Διαφορά με C++: απαγορεύεται σε ένα block η δήλωση μεταβλητής με ίδιο όνομα με μεταβλητή ενός πρόγονου block.

BlockInBlock.java



Διάρκεια Ζωής Μεταβλητής

- Διάρκεια ζωής μεταβλητής.
 - Χρονικό διάστημα ύπαρξης μιας μεταβλητής.
- Διάρκεια ζωής ανά είδος μεταβλητής.
 - Η διάρκεια ζωής ενός πεδίου είναι ίδια με τη διάρκεια ζωής του αντικειμένου στο οποίο ανήκει.
 - Η διάρκεια ζωής μιας παραμέτρου ή μιας τοπικής μεταβλητής είναι ίδια με τη διάρκεια της κλήσης της μεθόδου ή του κατασκευαστή που ανήκουν.



Μέθοδοι (1/3)

- Μια μέθοδος προαιρετικά μπορεί να επιστρέψει μια τιμή με την εντολή **return**.
- Ο τύπος της επιστρεφόμενης τιμής δηλώνεται στην επικεφαλίδα, πριν το όνομα της μεθόδου, αλλιώς δηλώνεται η λέξη **void**.
- Η εντολή **return** μπορεί να χρησιμοποιηθεί και χωρίς επιστρεφόμενη τιμή σε μεθόδους που δεν επιστρέφουν τιμή, οπότε και τερματίζεται η εκτέλεση της μεθόδου.



Μέθοδοι (2/3)

- Κλήση μεθόδων.
 - Οι μέθοδοι μπορούν να κληθούν από άλλες μεθόδους του ίδιου (**εσωτερική κλήση**) ή άλλου (**εξωτερική κλήση**) αντικειμένου.
- Εσωτερική κλήση.
 - Απλά με το όνομα της μεθόδου.
- Εξωτερική κλήση.
 - Με χρήση του τελεστή τελεία.
 - Π.χ. `student1.getAEM();`



Μέθοδοι (3/3)

- Μέθοδοι **προσπέλασης** (accessors).
 - Επιστρέφουν πληροφορίες για την κατάσταση ενός αντικειμένου.
 - Κάνουν χρήση της εντολής return.
- Μέθοδοι **μετάλλαξης** (mutators).
 - Αλλάζουν την κατάσταση ενός αντικειμένου.
 - Συνήθως είναι τύπου void.



Παραδείγματα Μεθόδων

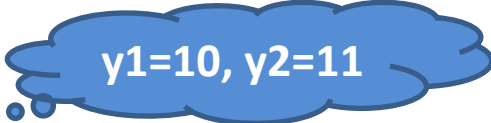
```
public int getAge() {  
    return age;  
}
```

```
public void setMarried(boolean b) {  
    married = b;  
}
```

```
public void printMessage() {  
    System.out.println("Καλημέρα!");  
}
```



Αριθμητικοί Τελεστές

- `+`, `-`, `*`, `/`, `%`, `++`, `--`
 - Εφαρμόζονται και σε εκφράσεις τύπου *char*.
 - Ο τελεστής `-` αφορά στην αφαίρεση, αλλά και ως μοναδιαίος στην αντιστροφή του πρόσημου.
 - Ο τελεστής `%` βρίσκει το υπόλοιπο και για αριθμούς κινητής υποδιαστολής.
 - `x1=10; x2=10; y1=x1++; y2=++x2;` 
- Συνδυασμός δυαδικών τελεστών με ανάθεση.
 - `counter += 5; //counter=counter+5;`

ArithmeticOperators.java



Κώδικας TicketMachinePlain (3/3)

```
public void insertMoney(int money) {  
    total += money;  
}
```

```
public void printTicket() {  
    System.out.println("Εισιτήριο " + price +  
        " λεπτών.");  
}
```



Κώδικας TicketMachinePlainApp

```
public class TicketMachinePlainApp
{

    public static void main(String[] args) {
        TicketMachinePlain tmp1;
        tmp1 = new TicketMachinePlain(60);
        tmp1.insertMoney(100);
        tmp1.printTicket();
    }
}
```



Τύποι Δεδομένων και Μνήμη (1/3)

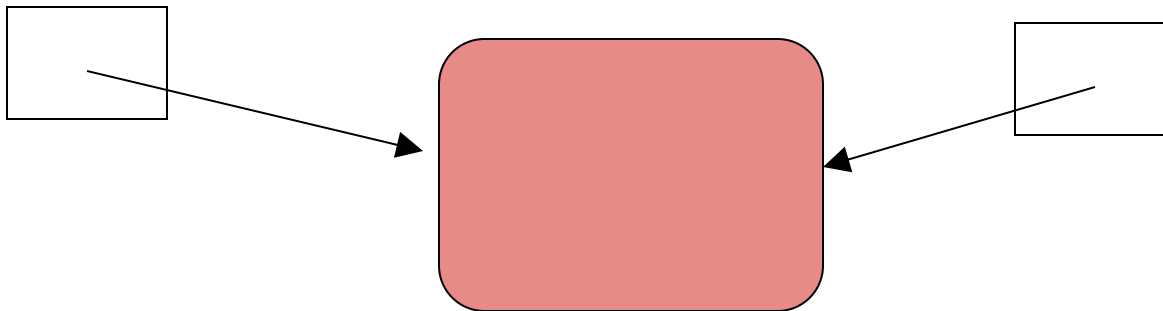
- Θεμελιώδεις τύποι.
 - Τοπικές μεταβλητές μεθόδων και τυπικές παράμετροι αποθηκεύονται στη στοίβα.
- Τύποι Αναφοράς.
 - Δυναμική δέσμευση μνήμης στο σωρό με **new**.
 - Αποθήκευση αναφοράς (δείκτη) στη θέση μνήμης.
 - Αντίθετα με τις C/C++ δεν μπορεί να τις πειράξει ο προγραμματιστής και δεν υπάρχουν τελεστές **&**, *****
 - Αυτόματη απελευθέρωση (garbage collector).



Τύποι Δεδομένων και Μνήμη (2/3)

`ObjectType a;`

`ObjectType b;`



`b = a;`

`int a;`



`int b;`



Τύποι Δεδομένων και Μνήμη (3/3)

- `int a, b;`
`a = 32;`
`b = a;`
`a = a + 1;`
`System.out.println(b);`
- `Person a, b;`
`a = new Person("Κώστας");`
`b = a;`
`a.changeName("Γιάννης");`
`System.out.println(b.getName());`

PrimitiveReference.java, Person.java



Παράδειγμα Κλάσης 2

- Αυτόματος πωλητής εισιτηρίων (βελτιωμένος).
 - Έλεγχος για εισαγωγή θετικού ποσού χρημάτων.
 - Έλεγχος αν έχει εισαχθεί το κατάλληλο ποσό.
 - Επιστροφή της διαφοράς χρημάτων.



Τελεστές Σύγκρισης

- ==, !=, <, >, <=, >=
 - Οι τελεστές ">", "<", "<=", ">=" εφαρμόζονται μόνο σε αριθμητικούς τύπους και *char*.
 - Το αποτέλεσμα τους είναι μια τιμή τύπου *boolean*.
 - Οι τελεστές "==", "!=" όταν εφαρμόζονται σε τύπους αναφοράς ελέγχουν μόνο αν οι μεταβλητές δείχνουν στην ίδια θέση μνήμης και όχι αν περιέχουν την ίδια πληροφορία.



Η Εντολή Ελέγχου if

```
if (έκφραση) {  
    // κώδικας  
}
```

boolean

```
if (έκφραση) {  
    // κώδικας  
} else {  
    // κώδικας  
}
```

```
if (έκφραση) {  
    // κώδικας  
} else if (έκφραση)  
{  
    // κώδικας  
} else if (έκφραση)  
{  
    // κώδικας  
} else {  
    // κώδικας  
}
```

If.java



Η Εντολή Ελέγχου Switch

char, byte, short, int, String (Java 7)

```
switch (έκφραση) {  
    case τιμή1:  
        // ακολουθία προτάσεων  
        break;  
    case τιμή2:  
        // ακολουθία προτάσεων  
        break;  
    ...  
    default:  
        // ακολουθία προτάσεων  
}
```



Ο Τριαδικός Τελεστής ?:

- έκφραση ? έκφραση : έκφραση

```
int val = -12;
```

```
int abs;
```

```
abs = val < 0 ? -val : val;
```

```
if (val < 0) {
```

```
    abs = -val;
```

```
} else {
```

```
    abs = val;
```

```
}
```

Ternary.java



Κώδικας TicketMachine (1/4)

```
public class TicketMachine {  
  
    private int price;  
  
    private int total;  
  
    private int balance;  
  
  
    public TicketMachine(int aPrice) {  
  
        price = aPrice;  
  
        total = 0;  
  
        balance = 0;  
  
    }  
  
}
```



Κώδικας TicketMachine (2/4)

```
public void insertMoney(int money) {  
    if (money > 0) {  
        balance += money;  
    } else {  
        System.out.println("Αρνητικά χρήματα!");  
    }  
}
```



Κώδικας TicketMachine (3/4)

```
public void printTicket() {  
    if (balance >= price) {  
        System.out.println("Εισιτήριο " +  
                             price + " λεπτών.");  
        total += price;  
        balance -= price;  
    } else {  
        System.out.println("Βάλε επιπλέον " +  
                             (price - balance) +  
                             " λεπτά.");  
    }  
}
```



Κώδικας TicketMachine (4/4)

```
public int refundBalance() {  
    int amountToRefund;  
    amountToRefund = balance;  
    balance = 0;  
    return amountToRefund;  
}
```



Κώδικας TicketMachineApp

```
public class TicketMachineApp {  
    public static void main(String[] args) {  
        TicketMachine tm1;  
        tm1 = new TicketMachine(60);  
        tm1.insertMoney(100);  
        tm1.printTicket();  
        int refund = tm1.refundBalance();  
        System.out.println("Ρέστα: " + refund);  
    }  
}
```





Τέλος Ενότητας

Επεξεργασία: Εμμανουήλ Ρήγας
Θεσσαλονίκη, Εαρινό Εξάμηνο 2013-2014



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