



Αντικειμενοστρεφής Προγραμματισμός

Ενότητα 6: Σχεδίαση Κλάσεων

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



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
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



Σχεδίαση Κλάσεων



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

Εισαγωγή

- Εμπορικό λογισμικό.
 - Διαρκής συντήρηση (διόρθωση σφαλμάτων) και επέκταση (προσθήκη λειτουργιών).
 - “*change or die*”.
- Καλοσχεδιασμένο λογισμικό.
 - Ο κώδικας πρέπει να διευκολύνει τις αλλαγές.
 - Βραχυπρόθεσμα αυτό απαιτεί προσπάθεια, όμως μακροπρόθεσμα μας σώζει από πολύ κόπο.
 - Δεν αφορά στην ορθή λειτουργία ή την τεκμηρίωση.



Σύζευξη

- Η έννοια της **σύζευξης** (coupling).
 - Αφορά στο βαθμό σύνδεσης των κλάσεων.
- Καλή σχεδίαση = χαλαρή σύζευξη.
 - Κάθε κλάση είναι ανεξάρτητη και επικοινωνεί μέσω μιας μικρής και καλά καθορισμένης διασύνδεσης.
- Πλεονεκτήματα.
 - Ευκολία αλλαγών στην υλοποίηση των κλάσεων.
 - Ευκολία συντήρησης και επέκτασης.



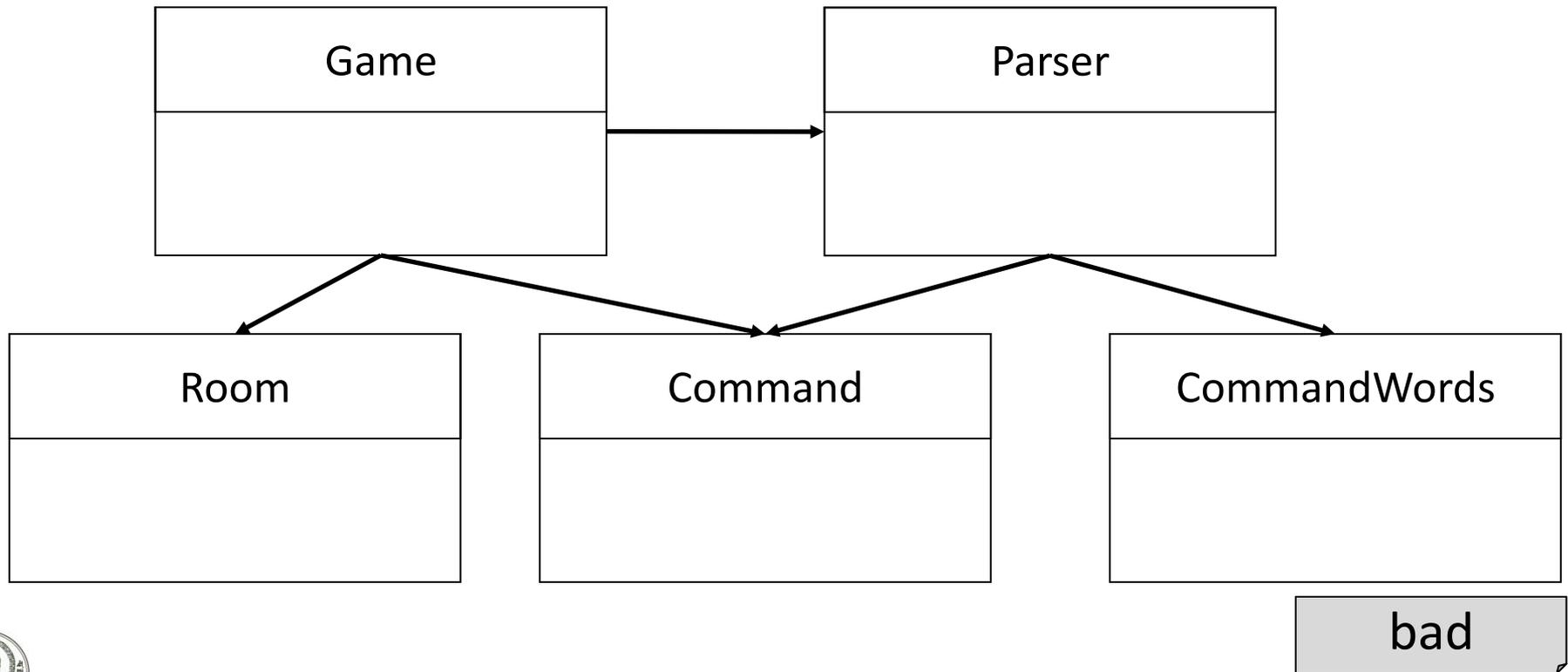
Συνοχή

- Η έννοια της **συνοχής** (cohesion).
 - Αφορά στον αριθμό και την ποικιλία των εργασιών μιας μονάδας κώδικα (μέθοδος, κλάση, πακέτο).
- Καλή σχεδίαση = υψηλή συνοχή.
 - Κάθε μονάδα κώδικα (μέθοδος, κλάση, πακέτο) να αφορά σε μία καλώς ορισμένη εργασία ή οντότητα.
- Πλεονεκτήματα.
 - Αναγνωσιμότητα (readability).
 - Επαναχρησιμοποίηση (reuse).
 - Ευκολία συντήρησης και επέκτασης.



Ένα Παιχνίδι Περιπέτειας

- Διάγραμμα κλάσεων.



Πανομοιότυπα Τμήματα Κώδικα

- Ένδειξη κακής σχεδίασης.
 - Μια αλλαγή θα πρέπει να γίνει σε πολλά σημεία, κάτι που μπορεί να διαφύγει της προσοχής μας.
- Σύμπτωμα χαμηλής συνοχής.
 - Ξεχωριστή μέθοδος για το τμήμα κώδικα που επαναλαμβάνεται (μεγαλύτερη συνοχή).
- Παράδειγμα.
 - Αφαιρέστε τις δύο εμφανίσεις πανομοιότυπου τμήματος κώδικα από την κλάση Game.
 - Στις μεθόδους printWelcome και goRoom.

better



Δύο Κανόνες Καλής Σχεδίασης

- Ενθυλάκωση / απόκρυψη πληροφορίας.
 - Η ενθυλάκωση χαλαρώνει τη σύζευξη.
- Σχεδίαση καθοδηγούμενη από αρμοδιότητες.
 - Κάθε κλάση πρέπει να είναι αρμόδια για τη διαχείριση των δικών της δεδομένων.
 - Χαλάρωση σύζευξης, αύξηση συνοχής.



Παράδειγμα (1/2)

- Παράδειγμα.
 - Επεκτείνετε το παιχνίδι με δύο επιπλέον εξόδους από ένα δωμάτιο (πάνω και κάτω).
- Ενθυλάκωση.
 - Χρήση ενθυλάκωσης για χαλάρωση της σύζευξης της κλάσης Room με την κλάση Game.
- Πρόβλεψη πιθανών αλλαγών.
 - Χρήση HashMap για αποθήκευση εξόδων .

better2, better3



Παράδειγμα (2/2)

- Σχεδίαση καθοδηγούμενη από αρμοδιότητες.
 - Η `printLocationInfo` της `Game` ασχολείται με τις εξόδους, οι οποίες όμως ανήκουν στην `Room`.
 - Κατασκευή μεθόδου `String getExitString()` στη `Room` και αντίστοιχη αλλαγή της `Game`.
 - Κατασκευή μεθόδου `String getLongDescription()` στη `Room` και αντίστοιχη αλλαγή της `Game`.

`better4, better5`



Έμμεση σύζευξη

- Έμμεση σύζευξη.
 - Μια κλάση βασίζεται σε εσωτερικές πληροφορίες μιας άλλης κλάσης χωρίς αυτό να είναι προφανές.
- Παράδειγμα.
 - Προσθήκη διαταγής look που εμφανίζει την περιγραφή του δωματίου.
 - Τι θα συμβεί αν δώσουμε τη διαταγή help;

better6, better7



Αλλαγές στον Κώδικα

- Τοπικότητα και πρόβλεψη αλλαγών.
 - Η πραγματοποίηση αλλαγών σε μια κλάση πρέπει να έχει ελάχιστες συνέπειες σε άλλες.
 - Ο καλός προγραμματιστής πρέπει να προβλέπει τις αλλαγές και να προνοεί για τη διευκόλυνση τους.
- Παράδειγμα.
 - Ο τρόπος αλληλεπίδραση με το χρήστη (κείμενο, GUI) να είναι αποκλειστική ευθύνη της Game.
 - Επιστροφή συμβολοσειράς με τις διαθέσιμες εντολές και εμφάνιση τους από τη Game.

better8



Τα Οφέλη της Συνοχής (1/2)

- Βελτιώνει την αναγνωσιμότητα του κώδικα.
 - Π.χ. μέθοδος `printWelcome` της `Game`.
- Αυξάνει την επαναχρησιμοποίηση του κώδικα.
 - Π.χ. η μέθοδος `getLongDescription` της `Room` χρησιμοποιείται σε τρεις μεθόδους της `Game`.
- Παράδειγμα.
 - Επέκταση του παιχνιδιού ώστε κάθε δωμάτιο να έχει ένα αντικείμενο (περιγραφή και βάρος).



Τα Οφέλη της Συνοχής (2/2)

- Προσθήκη ιδιοτήτων στην κλάση Room.
 - Κακή συνοχή, αφού πλέον η Room αφορά και αντικείμενα.
 - Προβλήματα σε επεκτάσεις όπως μεταφορά αντικειμένων από δωμάτιο σε δωμάτιο, πολλαπλά αντικείμενα σε κάθε δωμάτιο.
- Δημιουργία νέας κλάσης Item.

better9



Αναπαραγοντοθέτηση

- Αναπαραγοντοθέτηση (refactoring).
 - Αναδιάρθρωση μιας υπάρχουσας σχεδίασης για τη διατήρηση μιας καλής σχεδίασης κλάσεων όταν η εφαρμογή τροποποιείται ή επεκτείνεται .
- Η επέκταση γίνεται ιδανικά σε δύο βήματα.
 - Αναπαραγοντοθέτηση του κώδικα με στόχο την προετοιμασία των αλλαγών.
 - Αυξάνεται η ποιότητα του κώδικα.
 - Δεν αλλάζουμε ή επεκτείνουμε τη λειτουργικότητα.
 - Βελτίωση της λειτουργικότητας της εφαρμογής.



Παράδειγμα

- Επέκταση της εφαρμογής.
 - Ο παίκτης μπορεί να σηκώνει και να αφήνει αντικείμενα στο τρέχον δωμάτιο.
 - Ο παίκτης μπορεί να κουβαλάει οποιονδήποτε αριθμό αντικειμένων μέχρι ένα μέγιστο βάρος.
 - Ορισμένα αντικείμενα δεν μετακινούνται.
- Πως και που θα υλοποιηθούν όλα αυτά;
 - Τι πεδία χρειάζονται και σε ποιες κλάσεις θα προστεθούν;
 - Τι αλλαγές χρειάζεται η εφαρμογή;

good



Απαριθμητοί Τύποι

- Απαριθμητός τύπος (enumerated type).
 - Ορίζει ένα σύνολο από σταθερές.
 - Η δήλωση του γίνεται με τη λέξη κλειδί `enum`
- Μεταβλητές απαριθμητών τύπων.
 - Μπορούν να περιέχουν ως τιμή μία εκ των προκαθορισμένων σταθερών.
 - Πριν τη σταθερά πρέπει να προηγείται ο τύπος και ο τελεστής τελεία.



Παράδειγμα

```
public enum Day {  
    MONDAY, TUESDAY, WEDNESDAY, THURSDAY,  
    FRIDAY, SATURDAY, SUNDAY;  
}
```

```
public class Main {  
  
    public static void main(String[] args) {  
        Day d = new Day.MONDAY;  
    }  
}
```

enums.Day.java, enums.TestDay.java



Έλεγχος Τιμής με If και Switch

- Έλεγχος τιμής με if

```
if (day == Day.FRIDAY)
    System.out.println("Hurray!");
```

- Έλεγχος τιμής με switch

```
switch (day) {
    case FRIDAY:
        System.out.println("Hurray!");
        break;
    case MONDAY:
        System.out.println("Yuck!");
        break;
}
```



Απαριθμητός Τύπος ως Κλάση

- Στη Java ο απαριθμητός τύπος ορίζει κλάση.
 - Οι σταθερές είναι στην πραγματικότητα σταθερά (*final*) αντικείμενα αυτής της κλάσης.
 - Μπορούμε να ορίσουμε πεδία και μεθόδους.
 - Διατίθενται κάποιες έτοιμες μέθοδοι.
- Υπάρχουν όμως κάποιες διαφορές.
 - Δεν μπορούν να οριστούν νέα αντικείμενα με *new*.
 - Δεν επιτρέπονται *public* κατασκευαστές, όμως *default* κατασκευαστές μπορούν να χρησιμοποιηθούν κατά τον ορισμό σταθερών.



Έτοιμες Μέθοδοι

- **static enum-type[] values()**
 - Επιστρέφει πίνακα με όλες τις σταθερές.
- **static enum-type valueOf(String s)**
 - Επιστρέφει τη σταθερά που αντιστοιχεί στη συμβολοσειρά που δίνεται ως παράμετρος.
- **String toString()**
 - Επιστρέφει την σταθερά ως συμβολοσειρά.
- **int ordinal()**
 - Επιστρέφει ακέραιο με τη σειρά της σταθεράς.
- **int compareTo(enum-type e)**
 - Σύγκριση με άλλη σταθερά με βάση τη σειρά.



Παράδειγμα 1

```
Day all[] = Day.values();
for (Day day : all)
    System.out.println(day.toString() + " "
        + day.ordinal());
Scanner scanner = new Scanner(System.in);
System.out.print("Δώσε μια ημέρα: ");
Day day1 = Day.valueOf(scanner.nextLine());
System.out.print("Δώσε μια ακόμη ημέρα: ");
Day day2 = Day.valueOf(scanner.nextLine());
if (day1.compareTo(day2) < 0) {
    System.out.printf("Η %s προηγείται της
        %s%n", day1.toString(), day2.toString());
}
```

enums.TestDay.java



Παράδειγμα 2 (1/2)

```
public enum Planet {  
    VENUS    (4.869e+24, 6.0518e6),  
    EARTH    (5.976e+24, 6.37814e6);  
  
    private final double mass; // kilograms  
    private final double radius; // meters  
  
    Planet(double mass, double radius) {  
        this.mass = mass;  
        this.radius = radius;  
    }  
  
    public double surfaceGravity() {  
        return 6.67300E-11 * mass / (radius * radius);  
    }  
}
```

enums.Planet.java



Παράδειγμα 2 (2/2)

```
public double surfaceWeight(double otherMass) {  
    return otherMass * surfaceGravity();  
}
```

```
public static void main(String[] args) {  
    double g = Planet.EARTH.surfaceGravity();  
    double m = 78 / g;  
    for (Planet p : Planet.values())  
        System.out.printf("on %s weight is %f%n",  
            p.toString(), p.surfaceWeight(m));  
}
```

enums.Planet.java



Παράδειγμα Αναπαραγοντοθέτησης

- Ανεξαρτησία από τη γλώσσα.
 - Χρήση enumerations αντί για φυσική γλώσσα για αναπαράσταση των εντολών.

best





ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΙΚΑ
ΜΑΘΗΜΑΤΑ



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