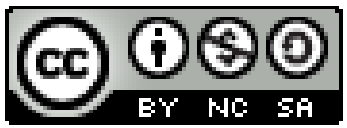
ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΙΚΑ
ΜΑΘΗΜΑΤΑ

Αντικειμενοστρεφής Προγραμματισμός

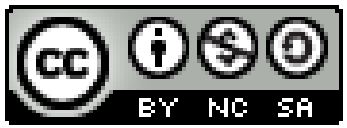
Ενότητα 15: Σχεδίαση Εφαρμογών

Γρηγόρης Τσουμάκας, Επικ. Καθηγητής
Τμήμα Πληροφορικής



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.

Σχεδίαση Εφαρμογών

Εισαγωγή

- Ανάλυση εφαρμογής.
  - Εντοπισμός και καταγραφή κλάσεων.
  - Εντοπισμός και καταγραφή αλληλεπιδράσεων μεταξύ κλάσεων.
- Σχεδίαση εφαρμογής.
  - Σχεδίαση των διασυνδέσεων των κλάσεων.

Η Μέθοδος Ρήμα/Ουσιαστικό

- Μέθοδος Ρήμα/Ουσιαστικό.
  - Καταγραφή των ουσιαστικών και των ρημάτων (ανά ουσιαστικό) στην περιγραφή της εφαρμογής.
- Ουσιαστικά.
  - Αναφέρονται σε «πράγματα».
  - Πιθανά αντικείμενα και κλάσεις.
- Ρήματα.
  - Αναφέρονται σε «δράσεις».
  - Πιθανές αλληλεπιδράσεις μεταξύ αντικειμένων, άρα πιθανές μέθοδοι των αντικειμένων.

Παράδειγμα (1/4)

Το σύστημα κρατήσεων θέσεων του κινηματογράφου πρέπει να αποθηκεύει κρατήσεις θέσεων για πολλές αίθουσες. Κάθε αίθουσα έχει θέσεις οργανωμένες σε σειρές. Οι πελάτες μπορούν να κάνουν κρατήσεις θέσεων, για τις οποίες τους δίνεται ένας αριθμός γραμμής και ένας αριθμός θέσης. Οι πελάτες μπορούν να ζητήσουν την κράτηση αρκετών γειτονικών θέσεων. Κάθε κράτηση αφορά μια συγκεκριμένη προβολή (μια συγκεκριμένη ταινία σε μια συγκεκριμένη ώρα). Οι προβολές γίνονται σε προκαθορισμένη ημερομηνία και ώρα, και σε μια ορισμένη αίθουσα προβολής. Το σύστημα αποθηκεύει τον αριθμό τηλεφώνου του πελάτη.

Παράδειγμα (2/4)

Το σύστημα κρατήσεων θέσεων του κινηματογράφου πρέπει να αποθηκεύει κρατήσεις θέσεων για πολλές αίθουσες. Κάθε αίθουσα έχει θέσεις οργανωμένες σε σειρές. Οι πελάτες μπορούν να κάνουν κρατήσεις θέσεων, για τις οποίες τους δίνεται ένας αριθμός γραμμής και ένας αριθμός θέσης. Οι πελάτες μπορούν να ζητήσουν την κράτηση αρκετών γειτονικών θέσεων. Κάθε κράτηση αφορά μια συγκεκριμένη προβολή (μια συγκεκριμένη ταινία σε μια συγκεκριμένη ώρα). Οι προβολές γίνονται σε προκαθορισμένη ημερομηνία και ώρα, και σε μια ορισμένη αίθουσα προβολής. Το σύστημα αποθηκεύει τον αριθμό τηλεφώνου του πελάτη.

Παράδειγμα (3/4)

Το σύστημα κρατήσεων θέσεων του κινηματογράφου πρέπει να αποθηκεύει κρατήσεις θέσεων για πολλές αίθουσες. Κάθε αίθουσα έχει θέσεις οργανωμένες σε σειρές. Οι πελάτες μπορούν να κάνουν κρατήσεις θέσεων, για τις οποίες τους δίνεται ένας αριθμός γραμμής και ένας αριθμός θέσης. Οι πελάτες μπορούν να ζητήσουν την κράτηση αρκετών γειτονικών θέσεων. Κάθε κράτηση αφορά μια συγκεκριμένη προβολή (μια συγκεκριμένη ταινία σε μια συγκεκριμένη ώρα). Οι προβολές γίνονται σε προκαθορισμένη ημερομηνία και ώρα, και σε μια ορισμένη αίθουσα προβολής. Το σύστημα αποθηκεύει τον αριθμό τηλεφώνου του πελάτη.

Παράδειγμα (4/4)

Σύστημα κρατήσεων θέσεων κινηματογράφου
- Αποθηκεύει (κρατήσεις θέσεων)
- Αποθηκεύει (αριθμό τηλεφώνου)

Πελάτης
- Κάνει κρατήσεις (θέσεων)
- Του δίνεται (αριθμός γραμμής, αριθμός θέσης)
- Ζητούν (κράτηση θέσεων)

Προβολή
- Γίνεται (ώρα, ημερομηνία, αίθουσα)

Αίθουσα
- Έχει (θέσεις)

Ταινία

Ημερομηνία

Ώρα

Θέση

Σειρά

Αριθμός θέσης

Αριθμος σειράς

Κράτηση θέσης

Κάρτες CRC

Όνομα κλάσης	Συνεργάτες
Αρμοδιότητες	

Σενάρια

- Σενάριο ή περίπτωση χρήσης (use case).
  - Μια δραστηριότητα που το σύστημα πρέπει να εκτελέσει ή να υποστηρίξει.
- Εφαρμογή σεναρίου (ομαδικά).
  - Κάθε μέλος της ομάδας αναλαμβάνει μια κλάση και περιγράφει τι κάνει η κλάση σε μια δεδομένη στιγμή στο πλαίσιο του σεναρίου.
- Ανακάλυψη και καταγραφή αλληλεπιδράσεων μεταξύ αντικειμένων (συνεργασίες).

Παραδείγματα

- Τηλεφωνική κράτηση 2 θέσεων για μια συγκεκριμένη ταινία σήμερα το βράδυ.
- Ένας πελάτης καλεί για να ακυρώσει μια κράτηση. Μπορεί να σας δώσει το όνομά του και την προβολή, αλλά έχει ξεχάσει τους αριθμούς θέσεων.
- Καλεί ένας πελάτης που έχει ήδη κάνει κράτηση. Θέλει να μάθει αν μπορεί να κρατήσει μια ακόμα θέση δίπλα στις υπόλοιπες.
- Το σύστημα πρέπει να διαμορφωθεί για έναν καινούργιο κινηματογράφο με δύο αίθουσες…
- Προγραμματίζεται για προβολή μια νέα ταινία…

Τα Σενάρια ως Εργαλείο Ανάλυσης

- Βοηθούν στο να διαπιστωθεί αν η περιγραφή του προβλήματος είναι πλήρης και ξεκάθαρη.
- Θα πρέπει να αφιερώνεται αρκετός χρόνος σε αυτήν τη διαδικασία ανάλυσης.
  - Ο εντοπισμός σφαλμάτων ή/και ελλείψεων σε αυτό το σημείο θα σώσει σημαντικό χαμένο κόπο αργότερα.

Σχεδίαση Κλάσεων

- Κάθε κάρτα CRC θα αντιστοιχεί σε μία κλάση.
- Οι συνεργάτες κάθε κλάσης
  - Αποκαλύπτουν την αλληλεπίδραση μεταξύ αντικειμένων των κλάσεων.
- Οι αρμοδιότητες κάθε κλάσης
  - Αποκαλύπτουν δημόσιες μεθόδους.
  - Και ορισμένες φορές και ιδιωτικά πεδία.

Σχεδίαση των Διασυνδέσεων

- Εφαρμογή των σεναρίων εκ νέου.
  - Αυτή τη φορά προσδιορίζοντας κλήσεις μεθόδων, παραμέτρους και επιστρεφόμενες τιμές.
- Καταγράφουμε τις υπογραφές των μεθόδων.
- Δημιουργούμε κλάσεις με κενές μεθόδους.
- Η προσεχτική σχεδίαση είναι σημείο κλειδί για μια πετυχημένη υλοποίηση.

Τεκμηρίωση Διασύνδεσης

- Συγγραφή σχολίων για κλάσεις και μεθόδους.
  - Περιγραφή του γενικότερου σκοπού τους.
- Συγγραφή σχολίων πριν την υλοποίηση.
  - Η έμφαση είναι στο «τι» και όχι στο «πως».
- Η σημαντικότητα της τεκμηρίωσης.
  - Γίνεται αισθητή σε έργα που αναπτύσσονται κατά τη διάρκεια πολλών μηνών ή/και ετών από πολλούς και διαφορετικούς προγραμματιστές.

Συνεργατική Ανάπτυξη

- Η ανάπτυξη γίνεται σε ομάδες.
  - Δεδομένης μια σωστής σχεδίασης, το λογισμικό θα έχει αναλυθεί σε πολλά αυτόνομα κομμάτια τα οποία μπορούν να ανατεθούν σε διαφορετικά άτομα ή ζεύγη προγραμματιστών.
- Pair programming.
  - Δύο προγραμματιστές παίρνουν εκ περιτροπής τους ρόλους του προγραμματιστή και του παρατηρητή.

Κατασκευή Πρωτοτύπου

- Πρωτότυπο.
  - Μια έκδοση της εφαρμογής στην οποία κάποια τμήματα προσομοιώνονται προκειμένου να εξετάσουμε άλλα τμήματα που έχουν αναπτυχθεί.
- Οφέλη.
  - Διερεύνηση μιας εφαρμογής από πολύ νωρίς.
  - Έγκαιρος εντοπισμός προβλημάτων σχεδίασης.
  - Αποφεύγεται η καθυστέρηση της ανάπτυξης λόγω εξαρτήσεων μεταξύ τμημάτων.

Εξέλιξη του Λογισμικού

- Μοντέλο καταρράκτη.
  - Ανάλυση, σχεδίαση, υλοποίηση, έλεγχος, παράδοση.
  - Δεν υποστηρίζει επιστροφή σε προηγούμενη φάση.
- Επαναληπτική ανάπτυξη.
  - Κατασκευή πρωτοτύπου από νωρίς.
  - Συχνή επικοινωνία με τον πελάτη.
  - Επαναληπτικά: ανάλυση, σχεδίαση, κατασκευή πρωτοτύπου, ανατροφοδότηση από τον πελάτη.
  - Σταδιακή, συνεχής εξέλιξη του λογισμικού.

Πρότυπα Σχεδίασης

- Οι σχέσεις μεταξύ των κλάσεων είναι σημαντικές και μπορεί να είναι πολύπλοκες.
- Ορισμένες σχέσεις εμφανίζονται συχνά σε διαφορετικές εφαρμογές.
- Τα πρότυπα σχεδίασης βοηθούν στο ξεκαθάρισμα των σχέσεων και στην επαναχρησιμοποίηση σχέσεων.

Δομή Προτύπου Σχεδίασης

- Όνομα του προτύπου.
- Πρόβλημα που αντιμετωπίζει.
- Λύση που παρέχει.
  - Δομές, συμμετέχοντες, συνεργασίες.
- Συνέπειες.
  - Αποτελέσματα, θετικά και αρνητικά στοιχεία.

Διακοσμητής (Decorator)

- Προσθήκη νέων λειτουργιών σε αντικείμενο.
- Ο διακοσμητής περιτυλίγει το αντικείμενο.
  - Έχει παρόμοια διεπαφή.
  - Μεταβιβάζει τις κλήσεις μεθόδων του στο αντικείμενο που περιτυλίγει …
  - … όμως προσθέτει ή/και παρεμβάλει επιπλέον λειτουργίες.
- Παράδειγμα: *java.io.BufferedReader*
  - Περιτυλίγει και επαυξάνει ένα *java.io.Reader.*

Παράδειγμα (1/4)

```
public interface Coffee {
   public abstract double getCost();
   public abstract String getIngredients();
}

public class SimpleCoffee implements Coffee {
   public double getCost() { return 1; }
   public String getIngredients() { return "Coffee"; }
}
```

Πηγή: Wikipedia

Παράδειγμα (2/4)

```
abstract class CoffeeDecorator implements Coffee {
  protected final Coffee decoratedCoffee;
  protected String ingredientSeparator = ", ";

  public CoffeeDecorator (Coffee decoratedCoffee) {
    this.decoratedCoffee = decoratedCoffee;
  }

  public double getCost() {
    return decoratedCoffee.getCost();
  }

  public String getIngredients() {
    return decoratedCoffee.getIngredients();
  }

}
```

Παράδειγμα (3/4)

```
public class Milk extends CoffeeDecorator {
 public Milk (Coffee decoratedCoffee) {
   super(decoratedCoffee);
 }

 public double getCost() {
   return super.getCost() + 0.5;

 }

 public String getIngredients() {
   return super.getIngredients() +
      ingredientSeparator + "Milk";

 }

}

public class Whip extends CoffeeDecorator

public class Sprinkles extends CoffeeDecorator
```

Παράδειγμα (4/4)

```
public class Main {


  public static final void main(String[] args) {
    Coffee c = new SimpleCoffee();
    c = new Milk(c);
    c = new Whip(c);
    c = new Sprinkles(c);
    System.out.println("Cost: " +
      c.getCost() +
      "Ingredients: " +
      c.getIngredients());
  }
```

Μοναδιαίο (Singleton)

- Εξασφαλίζει τη δημιουργία ενός και μόνο αντικειμένου από μία κλάση.
- Παραδείγματα από τη βιβλιοθήκη της Java.
  - Java.lang.Runtime
  - Java.awt.Toolkit
- Πως θα το υλοποιούσατε;

Early Loading / Eager Initialization

```
public class Singleton {
  private static Singleton instance = new Singleton();
  private Singleton() {}
  public static Singleton getInstance() {
   return instance;
  }
}
```

Lazy Initialization

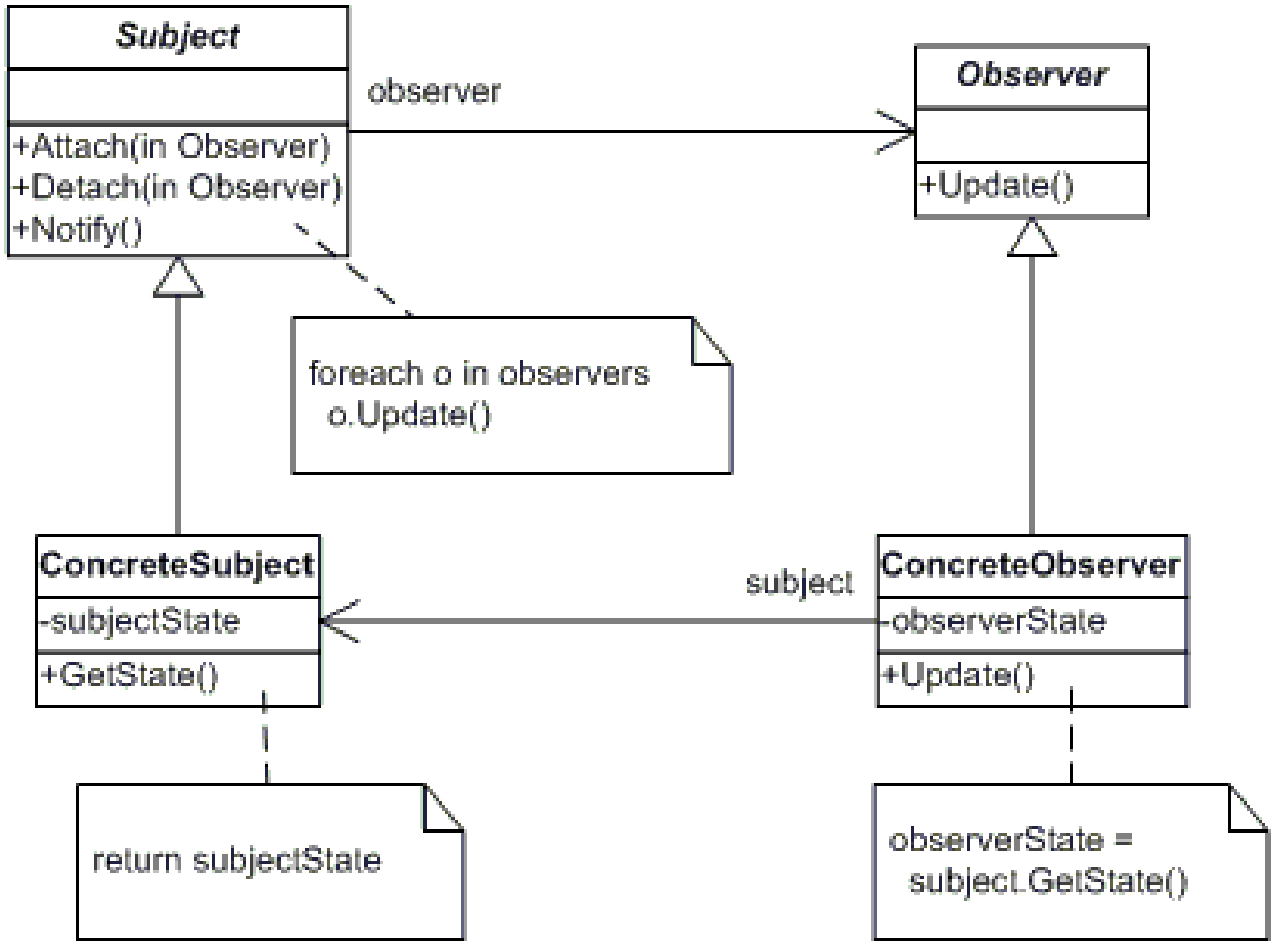
```
public class Singleton {
  private static Singleton instance = null;
  private Singleton() { }
  public static synchronized Singleton getInstance() {
    if (instance == null) {
      instance = new Singleton();
    }
    return instance;
  }
}
```

Το *synchronized* εξασφαλίζει σωστή λειτουργία σε εφαρμογές ταυτόχρονου προγραμματισμού

Παρατηρητής (Observer)

- Διαχωρισμός μοντέλου και όψης.
  - Διαχωρισμός της λογικής μιας εφαρμογής από τον τρόπο παρουσίασης της στην οθόνη.
- Γενικότερα ορίζει μια σχέση ένα-προς-πολλά.
  - Όταν η κατάσταση ενός αντικειμένου αλλάζει, άλλα αντικείμενα μπορούν να ενημερωθούν.
  - Αυτό επιτυγχάνεται με χαμηλή σύζευξη μεταξύ των παρατηρητών και του παρατηρούμενου.

Υλοποίηση Παρατηρητή



Παραδείγματα

- Κρεμάλα.
  - Άσκηση 1 εργαστηρίου 6.
- Τρίλιζα.
  - Άσκηση εργαστηρίου 7.
- Προσομοίωση «αλεπούδες και λαγοί».
  - Εναλλακτικές οπτικοποιήσεις πληθυσμών.

ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΙΚΑ
ΜΑΘΗΜΑΤΑ

Τέλος Ενότητας

Επεξεργασία: Εμμανουήλ Ρήγας

Θεσσαλονίκη, Εαρινό Εξάμηνο 2013-2014

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης