Υ325 - Αντικειμενοστρεφής Τεχνολογία

2021

Τμ. ΗΜ&ΤΥ - Πανεπιστήμιο Πατρών

Εβδομάδα 7 - 25/11/2021

Java GUI programming

Java GUI programming

Τα γραφικά στοιχεία (`Component`) είναι στοχεία όπως κουμπιά, κείμενο, πάνελ, μπάρες κλπ. Για κάθε στοιχείο υπάρχει η αντίστοιχη κλάση.

Τα γραφικά στοιχεία μπορούν να παράξουν γεγονότα, όταν αλλάξει η κατάστασή τους.

Για να εμφανιστούν τα στοιχεία στην οθόνη θα πρέπει να περιέχονται σε έναν κοντέινερ. Κοντέινερ είναι οι κλάσεις που επεκτείνουν την `Container`.

Java AWT vs Java Swing

Το Abstract Window Toolkit χρησιμοποιεί τις βιβλιοθήκες του λειτουργικού συστήματος για να δείξει τα γραφικά στοιχεία. Το AWT είναι ένα *"interface"* το οποίο δίνει πρόσβαση στις βιβλιοθήκες GUI του λειτουργικού συστήματος. Μια εφαρμογή AWT θα φαίνεται σαν μια κανονική εφαρμογή του λειτουργικού συστήματος. Συνεπώς η εφαρμογή Java μπορεί σε διαφορετικά λειτουργικά να φαίνεται διαφορορετική.

Το Java Swing έχει την ίδια όψη ανεξάρτητα από το λειτουργικό. Είναι όμως πιο "ελαφρύ" και παρέχει πιο ισχυρά γραφικά στοιχεία. Χρησιμοποιεί τις βιβλιοθήκες GUI του λειτουργικού μόνο για πολύ βασικά component, π.χ. για να κατασκευάσει ένα Frame.

Layout

Για να διατάξουμε τα στοιχεία μας σε μια γραφική διεπαφή, επιλέγουμε από μια σειρά από βοηθητικές κλάσεις. Όλες αυτές οι κληρονομούν από το interface `LayoutManager`.

Οι κλάσεις που υλοποιούν το `LayoutManager` παρέχουν διαφορετικές στρατηγικές για την τοποθέτηση των στοιχείων, ανάλογα με το είδος της διεπαφής που θέλουμε να σχεδιάσουμε.

Layout 2

Κάποιες από τις κλάσεις αυτές είναι

- `java.awt.BorderLayout`
- `java.awt.FlowLayout`
- `java.awt.GridLayout`
- `java.awt.CardLayout`
- `java.awt.GridBagLayout`
- `javax.swing.BoxLayout`
- `javax.swing.GroupLayout`
- `javax.swing.ScrollPaneLayout`
- `javax.swing.SpringLayout`

BorderLayout

Με τη στρατηγική 'BorderLayout', ο καμβάς μας χωρίζεται σε πέντε τμήματα (πάνω/κάτω/αριστερά/δεξιά/κέντρο):

- `public static final int NORTH`
- `public static final int SOUTH`
- `public static final int EAST`
- `public static final int WEST`
- `public static final int CENTER`

Συνεπώς μπορούμε να δείξουμε ταυτόχρονα 5 γραφικά στοιχεία στην περιοχή που διατάσσεται με 'BorderLayout'.

Παράδειγμα στις διαφάνειες της διάλεξης 7.

CardLayout

Στο `CardLayout` κάθε γραφικό στοιχείο είναι σαν μια “κάρτα” και εμφανίζει μόνο μία κάθε φορά.

Έχει δύο δημιουργούς:

- `CardLayout()`: *Creates a new card layout with gaps of size zero.*
- `CardLayout(int hgap, int vgap)`: *Creates a new card layout with the specified horizontal and vertical gaps.*

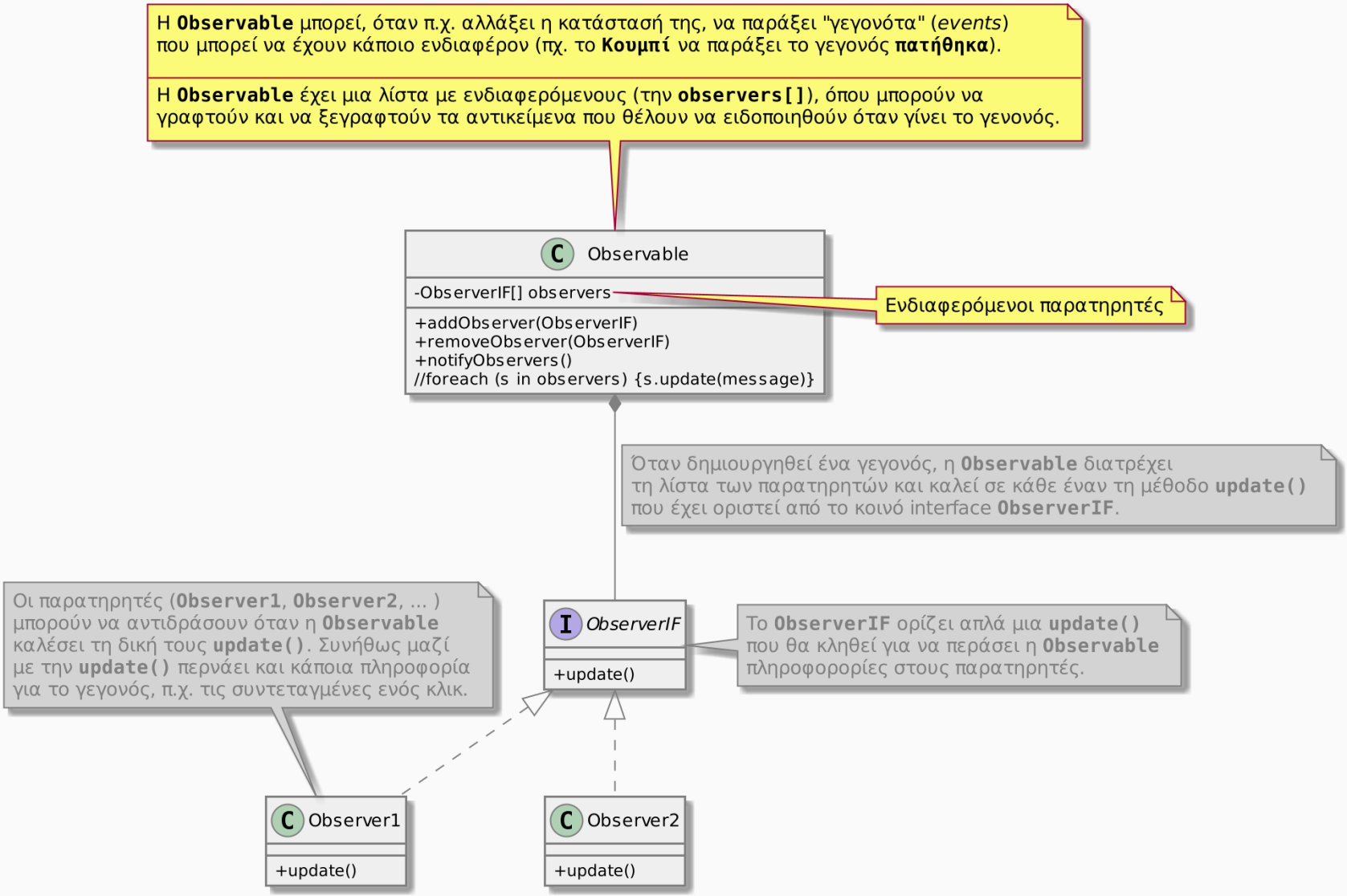
CardLayout 2

Χρήσιμες μέθοδοι:

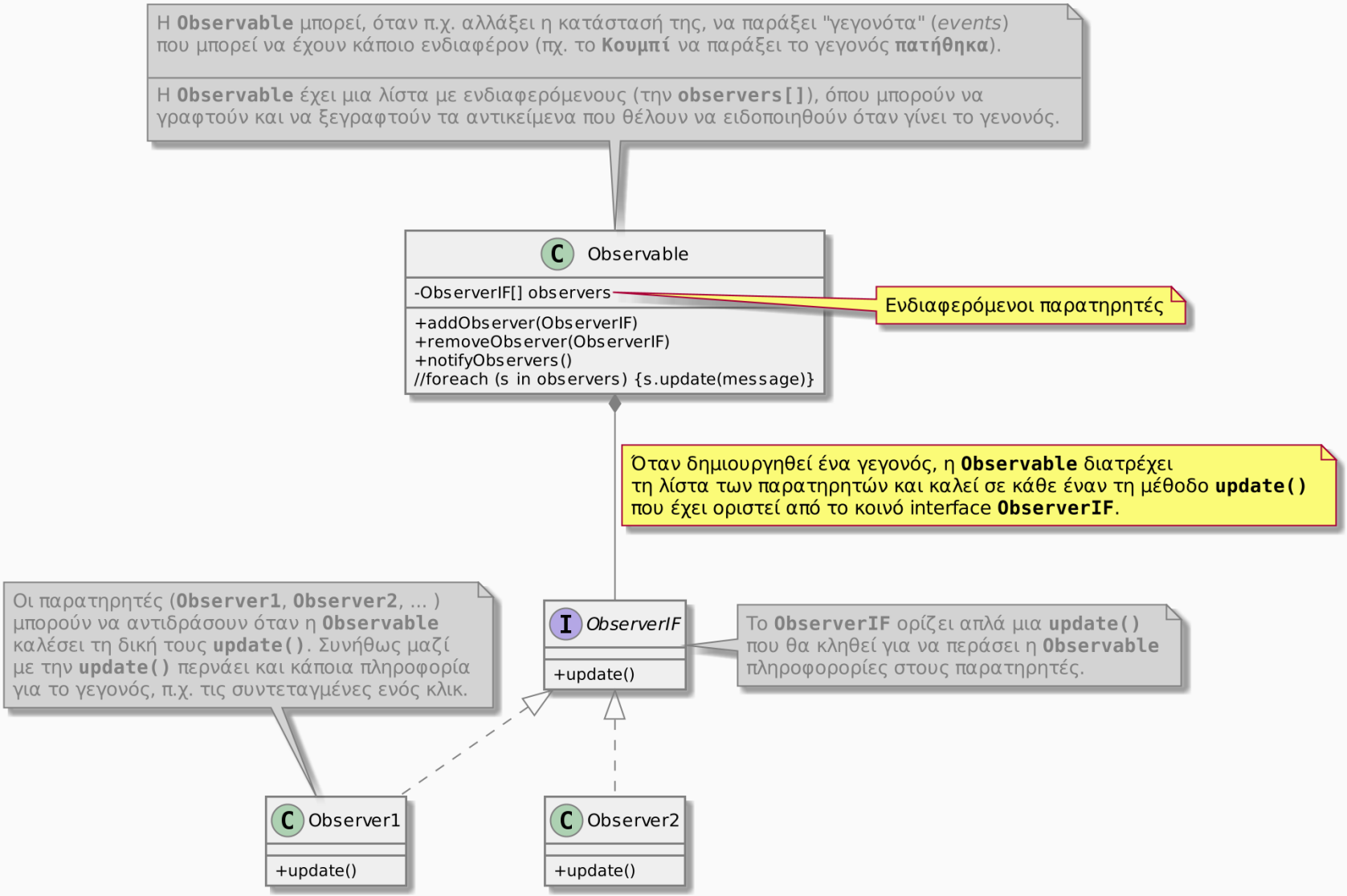
- `public void next(Container parent)`: *Flips to the next card of the specified container.*
- `public void previous(Container parent)`: *Flips to the previous card of the container.*
- `public void first(Container parent)`: *Flips to the first card of the container.*
- `public void last(Container parent)`: *Flips to the last card of the container.*

Χειρισμός γενονότων

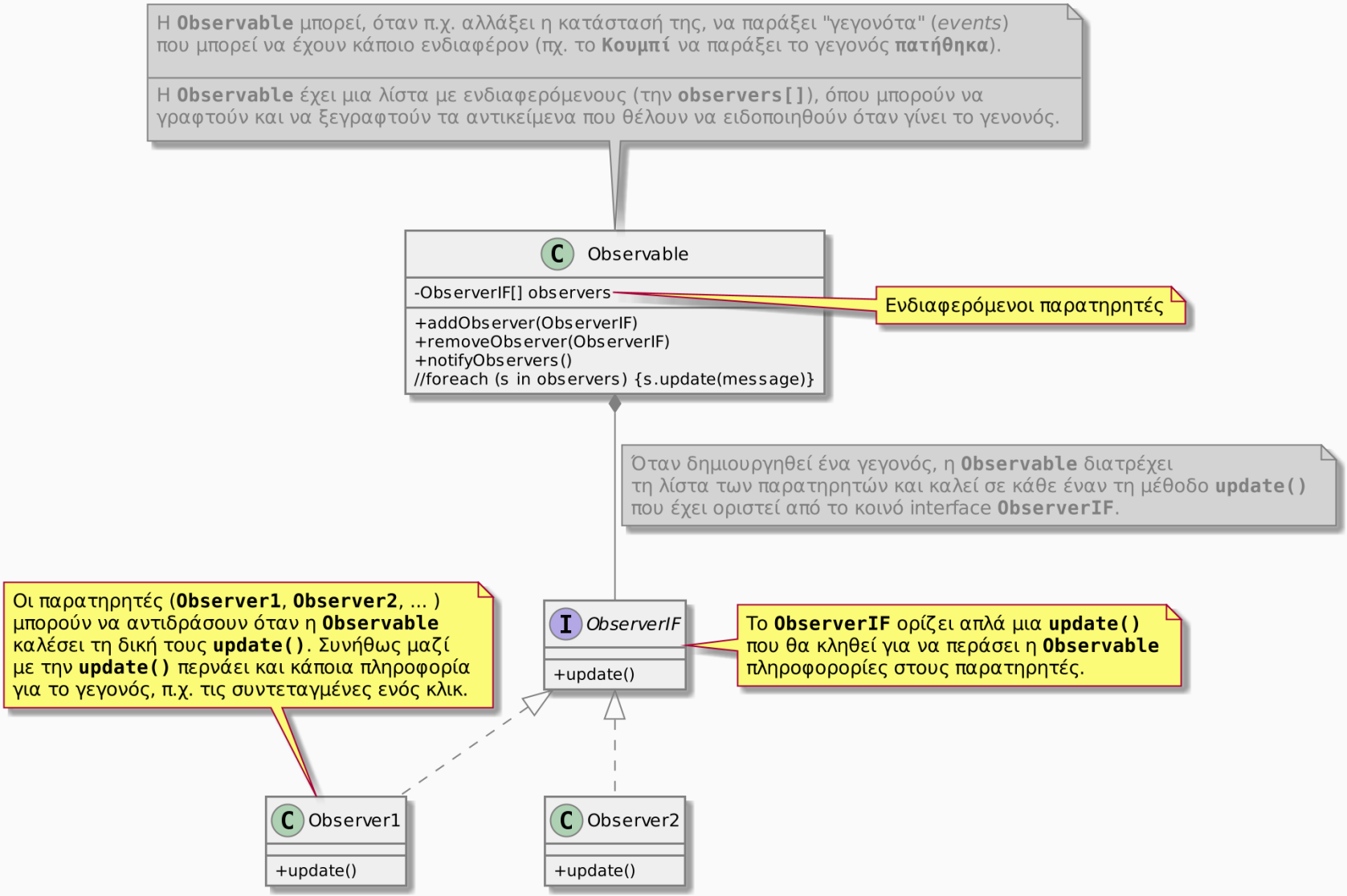
Χειρισμός γεγονότων 1



Χειρισμός γεγονότων 2



Χειρισμός γεγονότων 3



Χειρισμός γεγονότων 4

Στο Java AWT, ένα γραφικό στοιχείο π.χ. ένα `JButton`, μπορεί να δημιουργήσει ένα γεγονός (*event* ή *"action"* στην Java), και να πληροφορήσει όσες κλάσεις έχουν δηλώσει πως ενδιαφέρονται ότι συνέβει το γεγονός.

Τί είναι ένα action ή event;

- Το event είναι ένα αντικείμενο τύπου 'ActionEvent'. Το δημιουργεί το ίδιο το γραφικό στοιχείο.
- Ο δημιουργός του αντικειμένου τύπου 'ActionEvent' συμπεριλαμβάνει και κάποιες πληροφορίες (π.χ. ένα String, το πότε δημιουργήθηκε κλπ). Η 'ActionEvent' παρέχει μεθόδους που δίνουν πληροφορίες για το γεγονός, π.χ.:

```
public void actionPerformed(ActionEvent e) {
    System.out.println(e.paramString());
}
```

Χειρισμός γεγονότων 5

Ποιος θα ειδοποιηθεί όταν δημιουργηθεί το γεγονός;

- Τα αντικείμενα που γράφτηκαν σε μια λίστα που τηρεί το ίδιο το γραφικό στοιχείο.
- Τα αντικείμενα αυτά υλοποιούν το interface `ActionListener`.
- Δηλώνουμε πως θέλουμε να ενημερωθεί ένα αντικείμενο, καλούμε την μέθοδο `addActionListener(ActionListenter l)` του γραφικού αντικειμένου το οποίο μπορεί να παράξει γεγονότα:

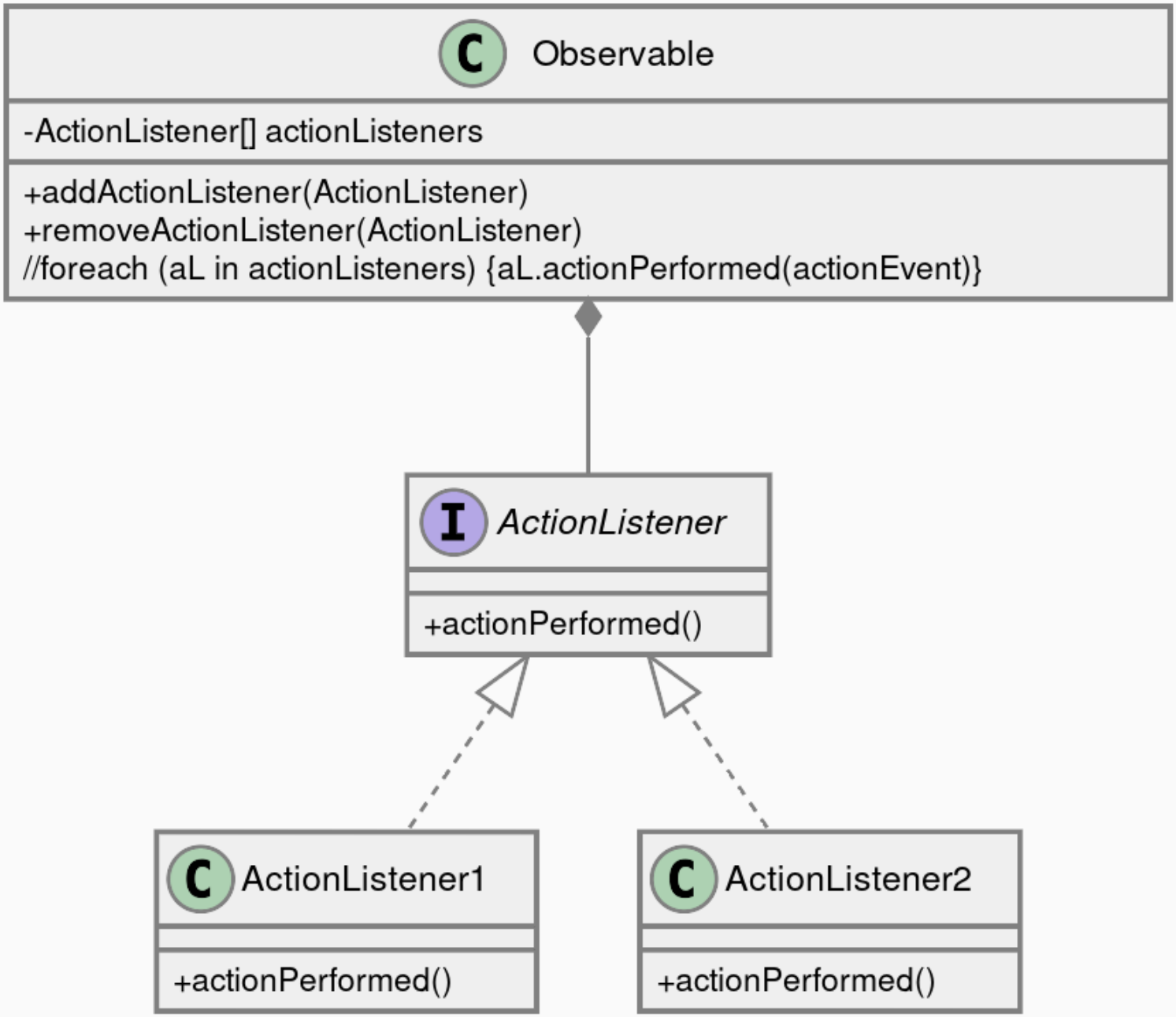
```
JButton jButton1 = new JButton("Πάτα με");
//Όταν το JButton1 παράξει γεγονός, θα ειδοποιηθεί το actionListerObj
jButton1.addActionListener(actionListenerObj);
```

Χειρισμός γεγονότων 6

Πως θα σταλεί η ειδοποίηση;

- Το γραφικό στοιχείο διατρέχει τη λίστα με τα αντικείμενα `ActionListerner` που έχουν δηλωθεί σε αυτό.
- Σε κάθε τέτοιο αντικείμενο καλεί τη μέθοδο `actionPerformed(ActionEvent e)` και περνάει το γεγονός τύπου 'ActionEvent' σαν παράμετρο.
    - Όλοι οι ακροατές του γραφικού στοιχείου έχουν υλοποιήσει τη μέθοδο `actionPerformed(ActionEvent e)` και περνάει το γεγονός τύπου 'ActionEvent', αφού υποχρεωτικά πρέπει να υλοποιήσουν το `ActionListener` interface.

Χειρισμός γεγονότων 7



Χειρισμός γεγονότων - ο ActionListener 1

Μπορούμε να γράψουμε τον κώδικα για τους `ActionListener`, τα αντικείμενα που θα ειδοποιηθούν για ένα γεγονός, με τρεις τρόπους:

1. να κάνουμε implement την ίδια μας την κλάση:

```
public class Example1 extends JFrame implements
    ActionListener {

    JButton jButton1 = new JButton("Πάτα με");
    //ο action listener είναι το ίδιο το αντικείμενο τύπου Example1
    jButton1.addActionListener(this);

    public void actionPerformed(ActionEvent e) {
        //ο χειρισμός του γεγονότος εδώ
    }
}
```

Χειρισμός γεγονότων - ο ActionListener 2

2. να φτιάξουμε μια άλλη κλάση:

```
public class Example1 extends JFrame  {
    JButton jButton1 = new JButton("Πάτα με");
    //ο action listener είναι το ίδιο το αντικείμενο τύπου Example1
    MyActionListener myActLstnr = new MyActionListener();
    jButton1.addActionListener(myActLstnr);

    //αφού θα χρησιμοποιήσουμε μόνο 1 φορά τον myActLstnr, μπορούμε
    //να δώσουμε ένα ανώνυμο αντικείμενο αντί για τις 2 πάνω γραμμές
    jButton1.addActionListener(new MyActionListener());
}

class MyActionListener implements ActionListener {
    public void actionPerformed(ActionEvent e) {
        //ο χειρισμός του γεγονότος εδώ
    }
}
```

Χειρισμός γεγονότων - ο ActionListener 3

3. σε μια ανώνυμη εσωτερική κλάση - anonymous inner class:

```
public class Example1 extends JFrame  {

    JButton jButton1 = new JButton("Πάτα με");
    //ο action listener είναι το ίδιο το αντικείμενο τύπου Example1
    jButton1.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            //ο χειρισμός του γεγονότος εδώ
        }
    });
}
```