



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΠΑΤΡΩΝ
UNIVERSITY OF PATRAS

ΑΝΟΙΚΤΑ ακαδημαϊκά
μαθήματα ΠΠ

Οντοκεντρικός Προγραμματισμός

Ενότητα 4: JAVA: ΔΙΕΠΑΦΕΣ ΧΡΗΣΤΗ, ΓΡΑΦΙΚΑ, APPLETS

Γραφικές Διεπαφές Χρήστη - Swing

ΔΙΔΑΣΚΟΝΤΕΣ: Ιωάννης Χατζηλυγερούδης, Χρήστος
Μακρής

Πολυτεχνική Σχολή

Τμήμα Μηχανικών Η/Υ & Πληροφορικής

ΓΡΑΦΙΚΕΣ ΔΙΕΠΑΦΕΣ ΧΡΗΣΤΗ

Εργαλειοθήκη SWING

ΓΡΑΦΙΚΕΣ ΔΙΕΠΑΦΕΣ ΧΡΗΣΤΗ-SWING (1)

Πλεονεκτήματα έναντι του awt

- Περισσότερα συστατικά
- Επεκτεταμένα χαρακτηριστικά συστατικών
- Καλύτερη εμφάνιση και αίσθηση
- Καλύτερος χειρισμός συμβάντων
- Πιο συμβατά προγράμματα στις διάφορες πλατφόρμες (συστατικά πλήρως υλοποιημένα σε java)



ΓΡΑΦΙΚΕΣ ΔΙΕΠΑΦΕΣ ΧΡΗΣΤΗ-SWING (2)

Ομοιότητες-Διαφορές από το awt

- Η βασική δομή του προγράμματος παραμένει.
- Οι βασικές έννοιες 'υποδοχέας', 'τομέας' 'συστατικό', και 'διαχειριστής διάταξης' παραμένουν.
- Η διαδικασία προσθήκης συστατικών διαφέρει.
- Η χρήση ενός συστατικού παραμένει ίδια.
- Αλλάζουν ελαφρώς τα ονόματα των κλάσεων (προσθήκη ενός «J» μπροστά συνήθως).
- Πακέτο: javax.swing



ΔΗΜΙΟΥΡΓΙΑ ΥΠΟΔΟΧΕΑ (1)

Διαδικασία (ίδια)

- Δημιουργία υποκλάσης της JFrame.
- Δημιουργία στιγμιότυπου της υποκλάσης στη μέθοδο main της υποκλάσης.

Δημιουργοί της JFrame

JFrame() → Δημιουργία παραθύρου χωρίς τίτλο

JFrame (String) → Δημιουργ. παραθ. με τίτλο

(Καλούνται μέσω του super απο τον δημιουργό της υποκλάσης)



ΔΗΜΙΟΥΡΓΙΑ ΥΠΟΔΟΧΕΑ (2)

Βασική δομή προγ/τος:

```
import java.awt.*;  
import javax.swing.*;
```

Συμπερίληψη εργαλειοθήκης

```
class MyApp extends JFrame {  
    public MyApp (String title) {  
        super(title);  
        <δημιουργία διαχειριστή διάταξης>  
        <δημιουργία-προσθήκη συστατικών>  
    }  
}
```

Δημιουργός

```
public static void main() {  
    MyApp app1 = new MyApp ("Application Window");  
    <διαχείριση παραθύρου>  
}
```

Δημιουργία
στιγμιότυπου



ΒΑΣΙΚΟΙ ΜΕΘΟΔΟΙ ΧΕΙΡΙΣΜΟΥ ΠΑΡΑΘΥΡΟΥ

pack() → μικρότερο δυνατό μέγεθος παραθ.

setVisible(boolean) → εμφάνιση/απόκρυψη παραθύρου

setBounds(int, int, int, int) → προσδιορισμός θέσης στην οθόνη (x, y, w, h)



ΔΙΑΧΕΙΡΙΣΤΕΣ ΔΙΑΤΑΞΗΣ

Καθορισμός Διαχειριστή Διάταξης

- Ισχύουν τα του awt
- Επιπλέον υπάρχει διαχειριστής εμφάνισης και αίσθησης



ΠΡΟΣΘΗΚΗ ΣΥΣΤΑΤΙΚΩΝ-ΤΟΜΕΑΣ

ΠΕΡΙΕΧΟΜΕΝΟΥ (1)

- Η προσθήκη συστατικών δεν γίνεται απ' ευθείας στον κύριο υποδοχέα, αλλά σ' έναν ενδιάμεσο υποδοχέα, που ονομάζεται τομέας περιεχομένου (content pane)
- Βασικά, ο κύριος υποδοχέας (πλαίσιο παραθύρου) αποτελείται από διάφορους τομείς. Ο κεντρικός τομέας είναι ο τομέας περιεχομένου



ΠΡΟΣΘΗΚΗ ΣΥΣΤΑΤΙΚΩΝ-ΤΟΜΕΑΣ

ΠΕΡΙΕΧΟΜΕΝΟΥ (2)

Διαδικασία

1. Δημιουργία ενός (στιγμιότυπου) τομέα (δημιουργός κλάσης JPanel).
2. Δημιουργία και προσθήκη συστατικών στον τομέα αυτό (`add(Component)`).
3. Προσθήκη του τομέα (χρήση `add`) ή καθορισμός του τομέα ως τομέα περιεχομένου (`setContentPane(Container)`)

```
JFrame f = new JFrame ();  
JButton b = new JButton ();  
Container contentPane = f.getContentPane ();  
contentPane.add(b );
```

Όλα τα συστατικά έχουν κοινή υπερκλάση την `JComponent`, απ' όπου κληρονομούν μεθόδους (`setEnabled`, `setVisible`, `setSize` κλπ)



ΕΤΙΚΕΤΕΣ

Κλάση: JLabel

Δημιουργοί:

```
JLabel (String)
```

```
JLabel (String, int)
```

(όπου int → SwingConstants.LEFT/CENTER/RIGHT)

```
JLabel (String, Icon, int)
```

Δημιουργία εικονιδίου

```
ImageIcon ic1 = new ImageIcon("icon1.gif");
```



ΠΛΗΚΤΡΑ/ΚΟΥΜΠΙΑ

Κλάση:

JButton

Δημιουργοί:

JButton (String)

JButton (Icon)

JButton (String, Icon)

Μέθοδοι:

setLabel (String)

getLabel ()



ΠΛΑΙΣΙΑ ΕΛΕΓΧΟΥ

Κλάση:

JCheckbox

Δημιουργοί:

JCheckbox (String)

JCheckbox (String, boolean)

JCheckbox (Icon)

JCheckbox (Icon, boolean)

JCheckbox (String, Icon)

JCheckbox (String, Icon, boolean)

Μέθοδοι:

setEnabled (boolean)



ΟΜΑΔΟΠΟΙΗΜΕΝΑ ΠΛΑΙΣΙΑ ΕΛΕΓΧΟΥ/ΚΟΥΜΠΙΑ ΕΠΙΛΟΓΗΣ

Κλάση: ButtonGroup

Δημιουργός: ButtonGroup()

Διαδικασία:

1. Δημιουργία στιγμιοτύπου ButtonGroup
2. Δημιουργία πλαισίων/κουμπιών
3. Προσθήκη πλαισίων/κουμπιών: add(Component)



ΠΑΡΑΔΕΙΓΜΑ

Βήμα 1

```
ButtonGroup lang = new ButtonGroup();
```

Βήμα 2

```
JCheckbox c1 = new JCheckbox ("Pascal", false);
```

```
JCheckbox c2 = new JCheckbox ("Java", false);
```

Βήμα 3

```
lang.add(c1);
```

```
lang.add(c2);
```



ΚΟΥΜΠΙΑ ΕΠΙΛΟΓΗΣ

Κλάση: JRadioButton

Δημιουργοί:

JRadioButton(String)

JRadioButton(String, boolean)

JRadioButton(Icon)

JRadioButton(Icon, boolean)

JRadioButton(String, Icon)

JRadioButton(String, Icon, boolean)

Μέθοδοι:

setEnabled(boolean)



ΛΙΣΤΕΣ ΕΠΙΛΟΓΗΣ (1)

Κλάση: JComboBox

Δημιουργός: JComboBox ()

Διαδικασία:

1. Δημιουργία σύνθετου πλαισίου

```
JComboBox langBox = new JComboBox();
```

2. Προσθήκη στοιχείων

```
langBox.addItem("Pascal");
```

3. Μετατροπή σύνθετου πλαισίου σε λίστα

```
setEditable(false);
```



ΛΙΣΤΕΣ ΕΠΙΛΟΓΗΣ (2)

Μέθοδοι :

getItem(int)

Επιστρέφει το στοιχείο στη θέση int (πρώτη θέση: 0)

getItemCount()

Επιστρέφει τον αριθμό των στοιχείων της λίστας

setSelectedIndex(int)

Επιλέγει το στοιχείο στη θέση int

getSelectedIndex()

Επιστρέφει τη θέση του τρέχοντος επιλεγμένου στοιχείου

getSelectedItem()

Επιστρέφει το όνομα του τρέχοντος επιλεγμένου στοιχείου



ΓΡΑΜΜΕΣ ΚΥΛΙΣΗΣ

Κλάση: JScrollBar

Δημιουργοί:

JScrollBar(int)

(int → SwingConstants.HORIZONTAL/VERTICAL)

JScrollBar(int, int, int, int, int)

(int → SwingConstants.HORIZONTAL/VERTICAL)

(int → αρχική τιμή)

(int → μέγεθος πλαισίου κύλισης)

(int → ελάχιστη τιμή)

(int → μέγιστη τιμή)

Επιλογή τιμής με ολίσθηση ενός πλαισίου.



ΠΕΔΙΑ ΚΕΙΜΕΝΟΥ (1)

Κλάση: JTextField, JPasswordField

Δημιουργοί:

JTextField(int)

JTextField(String, int)

JPasswordField(int)

JPasswordField(String, int)

(Χρήση της `setEchoChar(char)`)



ΠΕΡΙΟΧΕΣ ΚΕΙΜΕΝΟΥ

Κλάση:

TextArea

Δημιουργοί:

TextArea(int, int)

TextArea(String)

TextArea(String, int, int)



ΚΑΘΟΡΙΣΜΟΣ ΑΙΣΘΗΣΗΣ ΚΑΙ ΕΜΦΑΝΙΣΗΣ (1)

- Γίνεται μέσω της κλάσης **UIManager**
- Επιλογή από τρεις τύπους
 - Τύπος Windows
 - Τύπος Motif X-Windows
 - Τύπος Metal (java)
- Η επιλογή γίνεται από τη μέθοδο `setLookAndFeel (LookAndFeel)`
- Δημιουργία στιγμιοτύπου **LookAndFeel**:
 - (α) `getCrossPlatformLookAndFeelClassName ()`
(για επιλογή metal)
 - (β) `getSystemLookAndFeelClassName ()`
(για επιλογή αίσθησης & εμφάνισης του λειτουργικού συστήματος που χρησιμοποιείται)



ΚΑΘΟΡΙΣΜΟΣ ΑΙΣΘΗΣΗΣ ΚΑΙ ΕΜΦΑΝΙΣΗΣ

(2)

Η `setLookAndFeel` χρειάζεται χειρισμό εξαιρέσεων.

```
try {
    UIManager.setLookAndFeel(UIManager.getCrossPlatformLookAndFeelClassName());
}
catch (Exception e) {
    System.err.println("Can't set look and feel: "+e);
}
```



ΝΕΑ ΣΤΟΙΧΕΙΑ: ΠΡΟΤΥΠΑ ΠΑΡΑΘΥΡΑ ΔΙΑΛΟΓΟΥ

Κλάση: JOptionPane

Υπάρχουν τέσσερις τύποι παραθύρων διαλόγου

- Επιβεβαίωσης (ConfirmDialog)
- Εισόδου (InputDialog)
- Μηνύματος (MessageDialog)
- Επιλογής (OptionDialog)



ΝΕΑ ΣΤΟΙΧΕΙΑ: ΠΑΡΑΘΥΡΑ ΕΠΙΒΕΒΑΙΩΣΗΣ (1)

Ερώτηση με κουμπιά Yes, No, Cancel

Μέθοδος 1:

```
showConfirmDialog(Component, Object)
```

Component: ο υποδοχέας (null: κέντρο οθόνης)

Object: String, συστατικό ή Icon

Επιστρέφει: μία από τρεις ακέραιες τιμές:

(YES_OPTION, NO_OPTION, CANCEL_OPTION)

Π.χ.

```
int answer = JOptionPane.showConfirmDialog(null, "Are you sure to delete the file? ");
```



ΝΕΑ ΣΤΟΙΧΕΙΑ: ΠΑΡΑΘΥΡΑ ΕΠΙΒΕΒΑΙΩΣΗΣ (2)

Ερώτηση με κουμπιά Yes, No, Cancel

Μέθοδος 2:

```
showConfirmDialog(Component, Object, String, int, int)
```

String: κείμενο γραμμής τίτλου παραθύρου

int: YES_NO_CANCEL_OPTION, YES_NO_OPTION

int: ERROR_MESSAGE, INFORMATION_MESSAGE,
PLAIN_MESSAGE, QUESTION_MESSAGE, WARNING_MESSAGE

Π.χ.

```
int answer = JOptionPane.showConfirmDialog(null, "Error reading file", "File Input Error",  
JOptionPane.YES_NO_OPTION, JOptionPane.ERROR_MESSAGE);
```



ΝΕΑ ΣΤΟΙΧΕΙΑ: ΠΑΡΑΘΥΡΑ ΕΙΣΟΔΟΥ

Ερώτηση με πεδίο κειμένου για απάντηση

Μέθοδος 1:

`showInputDialog(Component, Object)`

Επιστρέφει: `String` (το κείμενο της απάντησης)

Π.χ.

```
string answer = JOptionPane.showInputDialog(null, "Enter your title:");
```

Μέθοδος 2:

`showInputDialog(Component, Object, String, int)`

Π.χ.

```
string answer = JOptionPane.showInputDialog(null, "Enter your title:",  
"Enter Title", JOptionPane. QUESTION_MESSAGE);
```



ΝΕΑ ΣΤΟΙΧΕΙΑ: ΠΑΡΑΘΥΡΑ ΜΗΝΥΜΑΤΟΣ

Εμφάνιση πληροφορίας

Μέθοδος 1:

```
showMessageDialog(Component, Object)
```

Επιστρέφει: Δεν επιστρέφει τιμή

Π.χ.

```
JOptionPane.showMessageDialog(null, "Title is missing");
```

Μέθοδος 2:

```
showMessageDialog(Component, Object, String, int)
```

Π.χ.

```
int answer = JOptionPane.showInputDialog(null, "The title is missing",  
    "Missing Component Message", JOptionPane.WARNING_MESSAGE);
```



ΝΕΑ ΣΤΟΙΧΕΙΑ: ΠΑΡΑΘΥΡΑ ΕΠΙΛΟΓΗΣ (1)

- Πιο περίπλοκο από τα άλλα
- Συνδυάζει χαρακτηριστικά όλων των άλλων

Μέθοδος:

```
showOptionDialog(Component, Object, String, int, int, Icon,  
                 Object[], Object)
```

Icon: στιγμιότυπο Icon αντί του υπάρχοντος

Object[]: πίνακας που περιέχει τα συστατικά που είναι οι επιλογές του παραθύρου.

Object: Η προεπιλεγμένη επιλογή, αν δεν χρησιμοποιηθούν τα YES_NO_CANCEL_OPTION, YES_NO_OPTION

Επιστρέφει: int



ΠΑΡΑΘΥΡΑ ΕΠΙΛΟΓΗΣ (2)

Π.χ.

```
Jbutton fav [] = new Jbutton[3];
fav[0] = new Jbutton("C++");
fav[1] = new Jbutton("Java");
fav[2] = new Jbutton("None");
int answer = JOptionPane.
    showInputDialog(null,
        "What is your favorite programming language?",
        "Favorite Language",
        0,
        JOptionPane.INFORMATION_MESSAGE,
        null,
        fav,
        fav[2]);
```



ΧΕΙΡΙΣΜΟΣ ΣΥΜΒΑΝΤΩΝ

- Μια κλάση που θέλει να αποκρίνεται σε συμβάντα πρέπει να υλοποιεί αντίστοιχη διεπαφή (interface), που λέγεται ακροατής συμβάντων (event listener).
- Κάθε ακροατής χειρίζεται ένα συγκεκριμένο είδος συμβάντος
- Μια κλάση μπορεί να υλοποιήσει όσους ακροατές χρειάζεται.

Διαδικασία

1. Υλοποίηση ακροατή συμβάντων
2. Συσχέτιση όποιων συστατικών επιθυμούμε με ένα ή περισσότερους ακροατές συμβάντων



ΒΑΣΙΚΟ ΠΛΑΙΣΙΟ ΠΡΟΓ/ΤΟΣ

```
import java.awt.GridLayout;
```

```
import javax.swing.*;
```

```
import java.awt.event;
```

Πακέτο χειρισμού συμβάντων.

```
class MyApp extends JFrame {
```

```
    public MyApp (String title) {
```

```
        super(title);
```

```
        <δημιουργία διαχειριστή διάταξης>
```

```
        <δημιουργία-προσθήκη συστατικών>} 
```

Κώδικας χειρισμού προτύπων συμβάντων: κλείνει την εφαρμογή όταν κλείνει το πλαίσιο (παράθυρο).

```
public static void main() {
```

```
    MyApp app = new MyApp ("Application Window");
```

```
    WindowListener L = new WindowAdapter() {
```

```
        public void windowClosing(WindowEvent e) {
```

```
            System.exit(0);}}
```

```
    app.addWindowListener(L);
```

```
    app.pack();
```

```
    app.setVisible(true);} }
```



ΑΚΡΟΑΤΕΣ ΣΥΜΒΑΝΤΩΝ (1)

- `ActionListener`

Συμβάντα ενέργειας: παράγονται από ενέργεια σε συστατικό, π.χ. πάτημα σε κουμπί

- `AdjustmentListener`

Συμβάντα ρύθμισης: παράγονται από ρύθμιση συστατικού π.χ. μετακίνηση γραμμής κύλισης

- `FocusListener`

Συμβάντα εστίασης: παράγονται από συστατικό που παίρνει ή χάνει την εστίαση, π.χ. πεδίο κειμένου

- `ItemListener`

Συμβάντα στοιχείου: παράγονται όταν ένα στοιχείο, π.χ. σε πλαίσιο ελέγχου, αλλάζει

- `KeyListener`

Συμβάντα πληκτρολογίου: παράγονται όταν εισάγουμε δεδομένα από το πληκτρολόγιο



ΑΚΡΟΑΤΕΣ ΣΥΜΒΑΝΤΩΝ (2)

- `MouseListener`
Συμβάντα ποντικιού: παράγονται από πατήματα του ποντικιού
- `MouseMotionListener`
Συμβάντα κίνησης ποντικιού: παράγονται από κίνηση του ποντικιού σε συστατικό)
- `TextListener`
Συμβάντα κειμένου: παράγονται από μεταβολές σε κείμενο
- `WindowListener`
Συμβάντα παραθύρου: παράγονται από μεταβολές σε παράθυρο, π.χ. ελαχιστοποίηση



ΣΥΣΧΕΤΙΣΗ ΑΚΡΟΑΤΩΝ ΣΥΜΒΑΝΤΩΝ ΜΕ ΣΥΣΤΑΤΙΚΑ (1)

- `addActionListener()` JButton, JCheckBox, JTextField, JRadioButton
- `addAdjustmentListener()` JScrollBar
- `addFocusListener()` όλα του Swing
- `addItemListener()` JButton, JCheckBox, JRadioButton
- `addKeyListener` όλα του Swing
- `addMouseListener` όλα του Swing
- `addMouseMotionListener` όλα του Swing
- `addTextListener` JTextField)
- `addWindowListwner` όλα των Jwindow, JFrame



ΣΥΣΧΕΤΙΣΗ ΑΚΡΟΑΤΩΝ ΣΥΜΒΑΝΤΩΝ ΜΕ ΣΥΣΤΑΤΙΚΑ (2)

Όλες οι μέθοδοι `add..` παίρνουν ένα όρισμα: το αντικείμενο που «ακούει» τα συμβάντα αυτού του είδους. Η χρήση του `this`, στην περίπτωση αυτή, δηλώνει την τρέχουσα κλάση ως ακροατή.

Π.χ.

```
Jbutton b = new Jbutton("button");  
b.addActionListener(this);
```

Μπορούμε να δηλώσουμε και ένα διαφορετικό αντικείμενο, αρκεί η κλάση του να υλοποιεί τη σωστή διεπαφή ακροατή.



ΜΕΘΟΔΟΙ ΧΕΙΡΙΣΜΟΥ ΣΥΜΒΑΝΤΩΝ

- Όταν συσχετίζουμε ένα ακροατή συμβάντων (διεπαφή) με μια κλάση, τότε η κλάση αυτή πρέπει να υλοποιεί όλες τις μεθόδους της αντίστοιχης διεπαφής.
- Κάθε τέτοια μέθοδος καλείται αυτόματα από το παραθυρικό σύστημα όταν συμβεί αντίστοιχο συμβάν.



ActionListener

ActionListener

Έχει μια και μοναδική μέθοδο, την `actionPerformed()`. Κάθε κλάση που υλοποιεί την `ActionListener` πρέπει να υλοποιεί την παρακάτω μέθοδο:

```
public void actionPerformed(ActionEvent e) {  
    <χειρισμός συμβάντος>  
}
```

Η `ActionEvent` είναι υποκλάση της `EventObject` (πακέτο `java.awt.event`)



getSource()

Μπορεί να χρησιμοποιηθεί για να προσδιορίσουμε το συστατικό στο οποίο στάλθηκε ένα συμβάν (ή με άλλα λόγια το συστατικό που «άκουσε» το συμβάν).

Π.χ.

```
public void actionPerformed(ActionEvent, e) {  
    Object comp = e.getSource();  
    if (comp == quitButton)  
        quitProgram();  
    else if (comp == sortRecords)  
        sortRecords();  
}
```



instanceof

Μπορεί να χρησιμοποιηθεί για να έλεγχο του είδους του συστατικού:

Π.χ.

```
public void actionPerformed(ActionEvent, e) {  
    Object comp = e.getSource();  
    if (comp instanceof JTextField)  
        calculateScore();  
    else if (comp instanceof JButton)  
        quitProgram();  
}
```



AdjustmentListener

Έχει μια και μοναδική μέθοδο, την

```
public void adjustmentValueChanged (AdjustmentEvent e) {  
    <χειρισμός συμβάντος>  
}
```



FocusListener

Μέθοδοι διασύνδεσης:

```
public void FocusGained(FocusEvent e) {  
    <χειρισμόςσυμβάντος>  
}
```

```
public void FocusLost(FocusEvent e) {  
    <χειρισμόςσυμβάντος>  
}
```



ItemListener

JButton, JCheckBox, JComboBox, JRadioButton

Όταν ένα στοιχείο επιλέγεται ή αποεπιλέγεται

Μέθοδος διασύνδεσης:

```
public void itemStateChanged(ItemEvent e) {  
<χειρισμόςσυμβάντος>  
}
```

Για να καθορίσετε το στοιχείο όπου συνέβη το συμβάν: μέθοδος getItem() στο αντικείμενο ItemEvent.



KeyListener

Μέθοδοι διασύνδεσης:

```
public void keyPressed(KeyEvent e) {  
    <χειρισμόςσυμβάντος>  
}  
public void keyReleased(KeyEvent e) {  
    <χειρισμόςσυμβάντος>  
}  
public void keyTyped(KeyEvent e) {  
    <χειρισμόςσυμβάντος>  
}
```

Χρήση μεθόδου getKeyChar() του KeyEvent



MouseListener

Μέθοδοι διασύνδεσης:

```
public void mouseClicked(MouseEvent e)
public void mouseEntered(MouseEvent e)
public void mouseExited(MouseEvent e)
public void mousePressed(MouseEvent e)
public void mouseReleased(MouseEvent e)
```

Χρήση μεθόδου `getPoint()` του `MouseEvent`



MouseEventListener

Μέθοδοι διασύνδεσης:

```
public void mouseDragged(MouseEvent e)
public void mouseMoved(MouseEvent e)
```

Χρήση μεθόδων του MouseEvent



WindowListener

Μέθοδοι διασύνδεσης:

```
public void windowActivated(WindowEvent e)
public void windowClosed(WindowEvent e)
public void windowClosing(WindowEvent e)
public void windowDeactivated(WindowEvent e)
public void windowDeiconified(WindowEvent e)
public void windowIconified(WindowEvent e)
public void windowOpened(WindowEvent e)
```



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στο πλαίσιο του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Αθηνών**» 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.



Σημείωμα Ιστορικού Εκδόσεων Έργου

Το παρόν έργο αποτελεί την έκδοση 1.0.1



Σημείωμα Αναφοράς

Copyright: Πανεπιστήμιον Πατρών, Ιωάννης Χατζηλυγερούδης, 2015.
«Οντοκεντρικός Προγραμματισμός». Έκδοση: 1.0.1 Πάτρα 2015. Διαθέσιμο
από τη δικτυακή διεύθυνση:

<https://eclass.upatras.gr/courses/CEID1105/>



Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.



Σημείωμα Χρήσης Έργων Τρίτων

