



ΕΚΠΑΙΔΕΥΤΙΚΟ ΕΓΧΕΙΡΙΔΙΟ

ΤΕΤΡΑΔΙΟ ΜΑΘΗΤΗ

Ανάδοχος Έργου



Κασταμονής 99α & Μακρυγιάννη
142 35 Ν. Ιωνία
τηλ. 210-2719100 fax 210-2718133
url : www.sdc.gr

Το παρόν εκπονήθηκε στο πλαίσιο
του Υποέργου 13 «Προσαρμογή Λογισμικού-Φάση III»
της Πράξης «Επαγγελματικό λογισμικό στην ΤΕΕ: επιμόρφωση και εφαρμογή»
(Γ' ΚΠΣ, ΕΠΕΑΕΚ, Μέτρο 2.3, Ενέργεια 2.3.2)

που συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση/Ευρωπαϊκό Κοινωνικό Ταμείο

Φορέας Υλοποίησης και Τελικός Δικαιούχος



Υπουργείο Εθνικής Παιδείας και Θρησκευμάτων
Ειδική Υπηρεσία Εφαρμογής Προγραμμάτων ΚΠΣ

Φορέας Λειτουργίας



Υπουργείο Εθνικής Παιδείας και Θρησκευμάτων
Διεύθυνση Σπουδών Δευτεροβάθμιας Εκπαίδευσης-Τμήμα Β'

Επιστημονικός Τεχνικός Σύμβουλος



Ερευνητικό Ακαδημαϊκό Ινστιτούτο Τεχνολογίας Υπολογιστών

Υπεύθυνος Πράξης

2003-2007 Προϊστάμενος Μονάδας Α1-Ειδική Υπηρεσία Εφαρμογής Προγραμμάτων ΚΠΣ-ΥΠΕΠΘ.
2007- Προϊστάμενος Μονάδας Α1β-Ειδική Υπηρεσία Εφαρμογής Προγραμμάτων ΚΠΣ-ΥΠΕΠΘ.



ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ ΕΠΕΑΕΚ



ΕΥΡΩΠΑΪΚΗ ΕΝΩΣΗ
ΣΥΧΡΗΜΑΤΟΔΟΤΗΣΗ
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Η ΠΑΙΔΕΙΑ ΣΤΗΝ ΚΟΡΥΦΗ
Επιχειρησιακό Πρόγραμμα
Εκπαίδευσης και Αρχικής
Επαγγελματικής Κατάρτισης

ΕΙΣΑΓΩΓΗ

Διαδικασία εκκίνησης της Java και το NetBeans IDE.

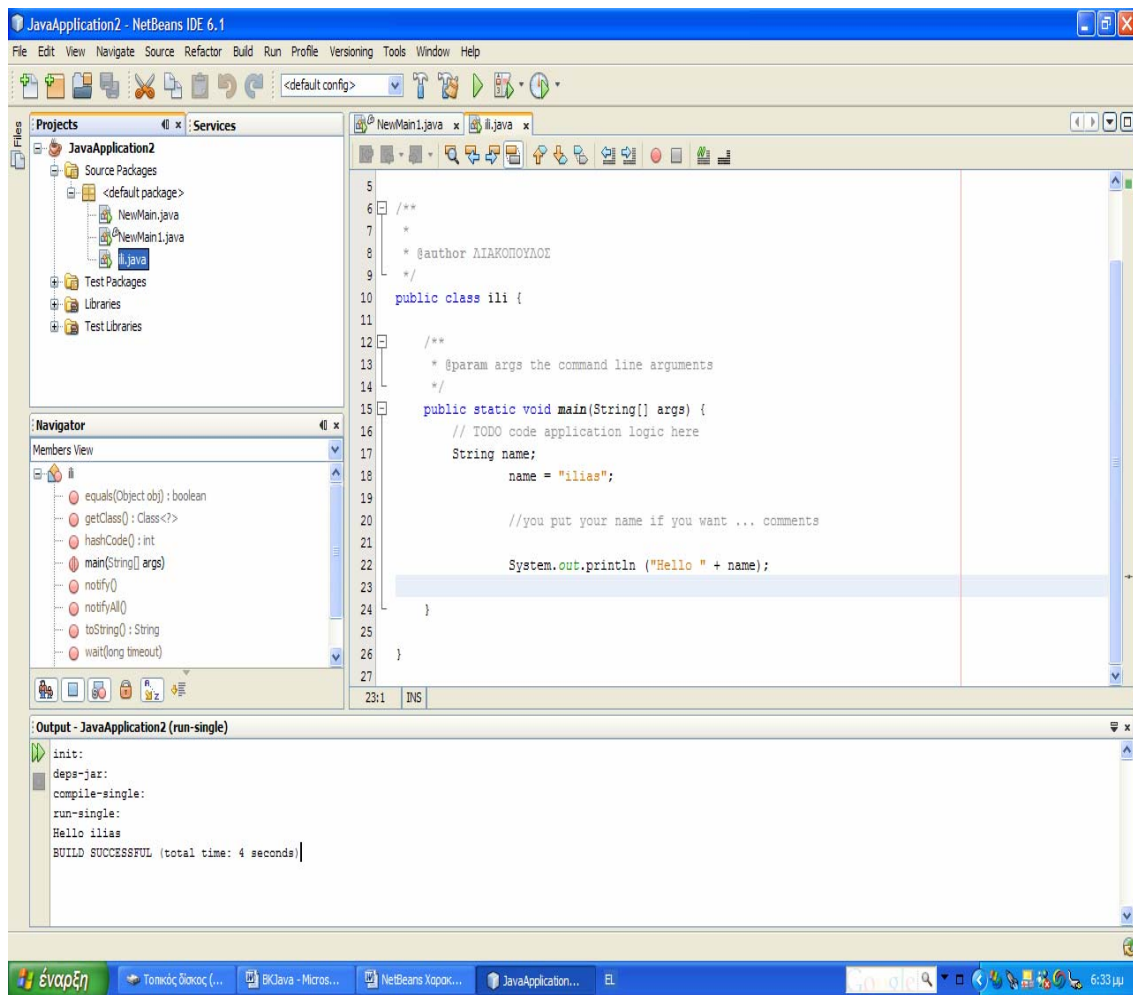
Για να μπορέσουμε να γράψουμε προγράμματα σε Java, χρειάζεται να διαθέτουμε το ειδικό πακέτο ανάπτυξης εφαρμογών (**Java development kit** ή **JDK**).

Το **JDK** είναι μία γλώσσα προγραμματισμού. Συγχρόνως περιέχει μεταγλωττιστή, διερμηνευτή, αποσφαλιματωτή (debugger), αποσυμβολομεταφραστή (disassembler).

Αρχικά θα πρέπει να κατεβάσουμε από τη σελίδα της Sun Microsystems το κατάλληλο JDK, το οποίο θα πρέπει να είναι έκδοση JDK 1.5.5_0 ή μεταγενέστερο, γιατί το NetBeans το απαιτεί. Η διαδικασία που θα ακολουθηθεί είναι :

Πηγαίνετε στη σελίδα της Sun → **http://sun.com** , μετά στη επιλογή downloads και από τη συγκεκριμένη σελίδα επιλέξτε να κατεβάσετε το JDK 6. Αφού ολοκληρωθεί το κατέβασμα, τότε εγκαταστήστε το JDK 6 στο υπολογιστή σας, ακολουθώντας βήμα - βήμα τη διαδικασία που προσδιορίζετε από τις οθόνες εγκατάστασης.

Αφού ολοκληρωθεί η εγκατάσταση του JDK 6, το επόμενο βήμα είναι να εγκαταστήσετε το NetBeans IDE που σας παρέχεται. Το περιβάλλον του NetBeans IDE φαίνεται παρακάτω :



Δημιουργία μιας εφαρμογής Java

Για να δημιουργηθεί μια εφαρμογή Java πρέπει να ακολουθηθούν τα εξής βήματα :

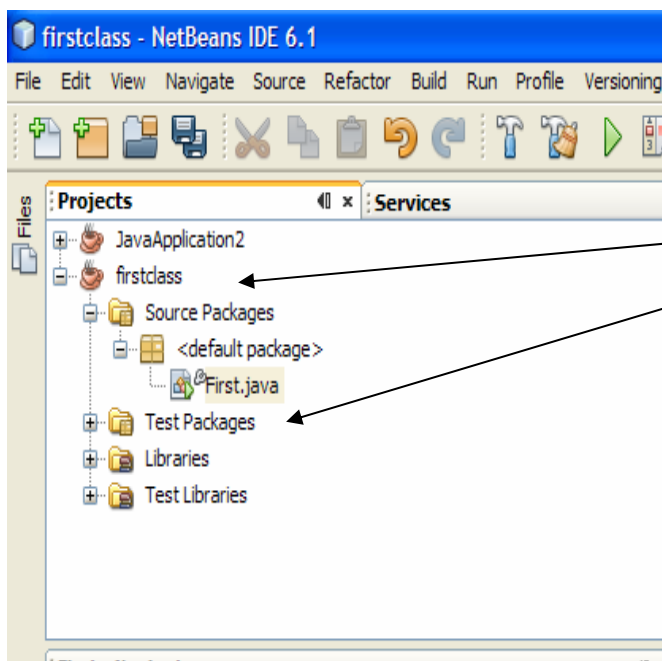
Αρχικά πρέπει να δημιουργηθεί ένα project το οποίο θα περιέχει όλα τα αρχεία (εφαρμογές) του project.

Για τη δημιουργία του project κάνετε τα εξής :

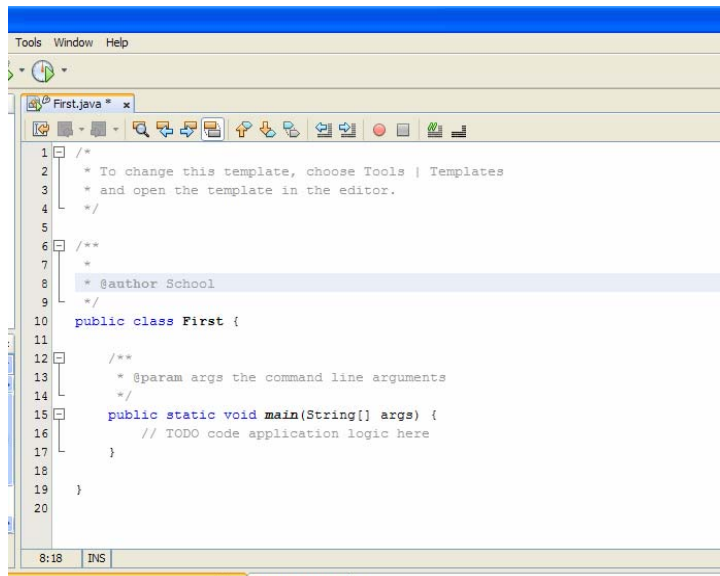
File new project → στη καρτέλα categories επιλέγεις Java → στη καρτέλα projects επιλέγεις Java Applications → κλικ Next. Στη επόμενη καρτέλα που ανοίγει επιλέγεις : Project Name, Project Location, Project Folder. Απενεργοποιείς τις επιλογές Create Main Class και Set as Main Project → κλικ στο Finish.

Αμέσως μετά πρέπει να δημιουργηθεί αρχείο της εφαρμογής μέσα στο project, το οποίο από μόνο του θα έχει προέκταση .java. Για να δημιουργηθεί κάνεις τα εξής :

File new File → στη καρτέλα που ανοίγει, στο project επιλέγεις το project στο οποίο θα καταχωρηθεί το αρχείο, στη επιλογή categories επιλέγεις Java, και στη καρτέλα file types επιλέγεις Java main Class. → κλικ Next . Στη επόμενη καρτέλα επιλέγεις όνομα για τη κλάση και μετά κλικ στο Finish. Αφού ολοκληρωθεί επιτυχώς η παραπάνω διαδικασία, έχετε τη παρακάτω οθόνη :



Βλέπετε αριστερά το project firstclass που δημιουργήθηκε, καθώς και τη εφαρμογή First.java.



```
1  /*
2  * To change this template, choose Tools | Templates
3  * and open the template in the editor.
4  */
5
6  /**
7  *
8  * @author School
9  */
10 public class First {
11
12     /**
13     * @param args the command line arguments
14     */
15     public static void main(String[] args) {
16         // TODO code application logic here
17     }
18
19 }
20
```

Βλέπετε αριστερά το αρχείο First.java, στο οποίο είμαστε έτοιμοι να πληκτρολογήσουμε τον κώδικα του προγράμματος.

Πληκτρολογούμε τη εντολή που ακολουθεί στο αρχείο First.java.

```
System.out.println (“Hello World!!!!!!!!”);
```

Πρέπει να δώσουμε προσοχή στη πληκτρολόγηση γιατί στη Java έχουν διαφορά τα κεφαλαία από τα πεζά. Επίσης, κάθε εντολή στη Java ολοκληρώνεται με το ερωτηματικό (;). Οπότε το αρχείο First.java διαμορφώνεται ως εξής :

```

1  /*
2  * To change this template, choose Tools | Templates
3  * and open the template in the editor.
4  */
5
6  /**
7   *
8   * @author School
9   */
10 public class First {
11
12     /**
13      * @param args the command line arguments
14      */
15     public static void main(String[] args) {
16         System.out.println("Hello World!!!!");
17     }
18
19 }
20

```

Στο σημείο αυτό είναι απαραίτητο να εξηγήσουμε ότι βλέπουμε στο ορισμό του προγράμματος.

Class First → Ορισμός της κλάσης First. Ο ορισμός περιλαμβάνει μόνο μία μέθοδο, τη main().

Οι αγκύλες {} → πηγαίνουν ανά ζεύγη και περικλείουν αυτόνομα κομμάτια κώδικα.

Public → Η λέξη public δηλώνει ότι η μέθοδος είναι προσπελάσιμη από παντού.

static → Η λέξη static δηλώνει ότι η μέθοδος είναι προσπελάσιμη ακόμη και αν δεν έχουν δημιουργηθεί αντικείμενα της κλάσης.

Void → σημαίνει ότι δεν επιστρέφει καμιά τιμή.

Main () → Η βασική μέθοδος.

Μέσα στη Main() υπάρχει η εντολή : System.out.println("Hello World!!");

System → Είναι το όνομα μίας βασικής κλάσης, η οποία περιλαμβάνει αντικείμενα και μεταβλητές, για υποστήριξη εισαγωγής δεδομένων από το

πληκτρολόγιο. Επίσης χρησιμοποιείται και για την έξοδο χαρακτήρων στην οθόνη.

out → Το αντικείμενο out δηλώνει σαν τυπική έξοδο την οθόνη και είναι μέλος της κλάσης System.

println () → Είναι μέθοδος του αντικειμένου out και τυπώνει στην οθόνη την φράση που βρίσκεται μέσα στην παρένθεση. Προσοχή στο ερωτηματικό (;), το οποίο πρέπει να είναι στο τέλος κάθε εντολής.

Μεταγλώττιση ενός προγράμματος

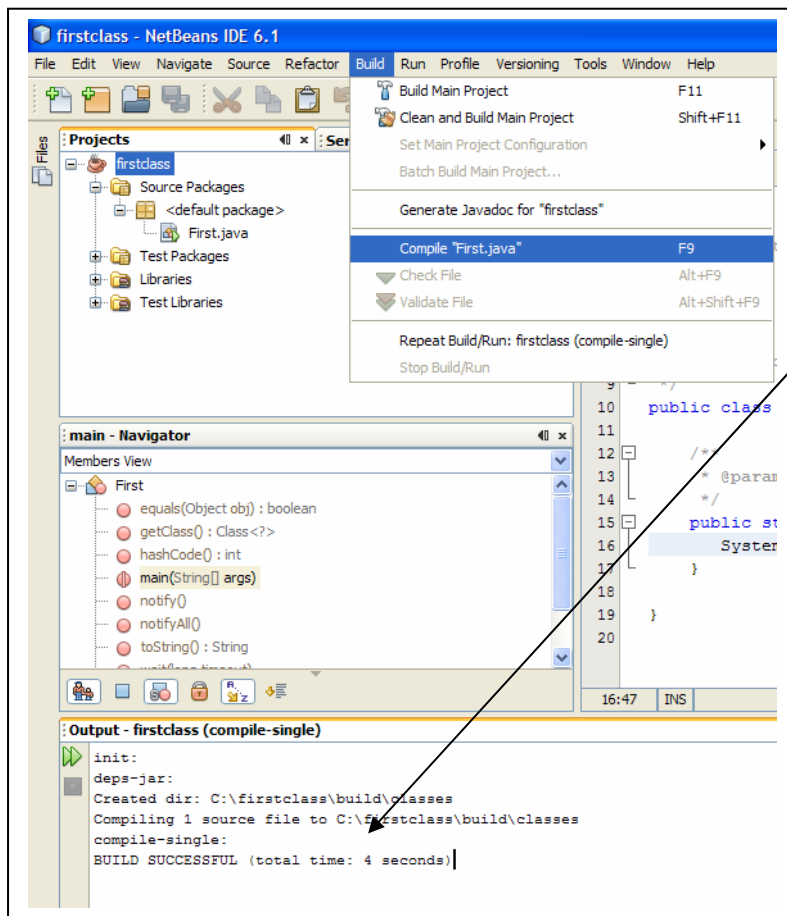
Τρόπος Α'

Χρησιμοποιώντας το NetBeans μεταγλωττίζουμε ένα πρόγραμμα με τη ακόλουθη διαδικασία :

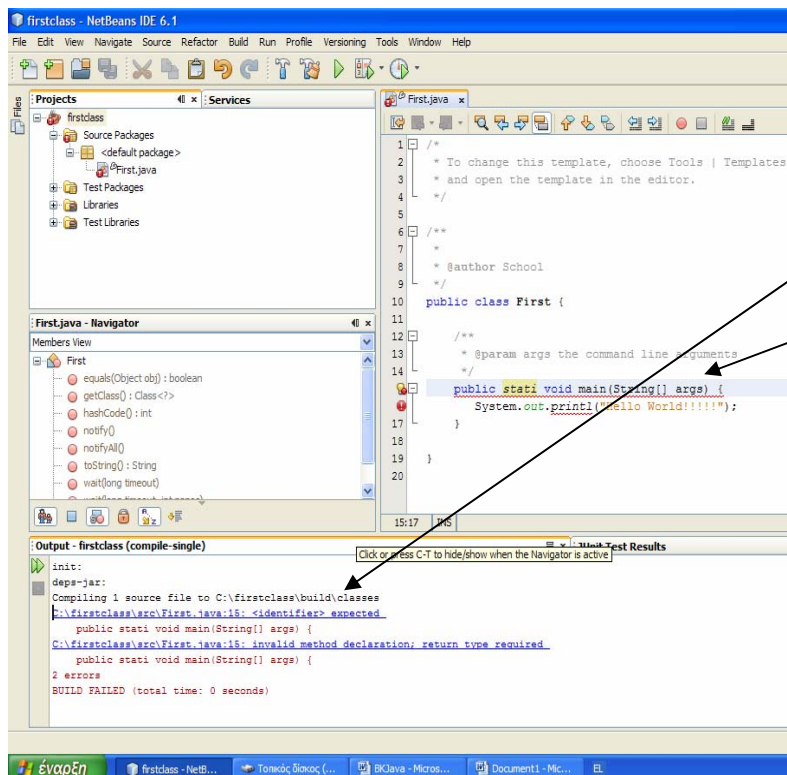
Στη γραμμή μενού → κλικ στη επιλογή Build → κλικ στη επιλογή Compile "First.java". Αν η διαδικασία είναι επιτυχής, δηλαδή δεν υπάρχουν συντακτικά λάθη, τότε στη οθόνη του NetBeans στη περιοχή output, στο κάτω μέρος της οθόνης, παίρνεις το μήνυμα BUILD SUCCESSFUL.

Αν υπάρχουν λάθη, τότε στη περιοχή του output εμφανίζεται ο συνολικός αριθμός των λαθών και συγχρόνως στη περιοχή του προγράμματος υπογραμμίζονται με κόκκινο χρώμα τα λάθη τα οποία εντοπίστηκαν.

Η παραπάνω διαδικασία φαίνεται στις παρακάτω οθόνες :



Διαδικασία επιτυχούς μεταγλώττισης του προγράμματος First.java.



Διαδικασία μεταγλώττισης με λάθη στο κώδικα του προγράμματος

Αν η διαδικασία μεταγλώττισης είναι επιτυχής τότε δημιουργείται το First.class αρχείο, που είναι ένα αρχείο που περιέχει κώδικα Byte.

Που βρίσκονται τα αρχεία First.java και First.class ;

Αν υποθέσουμε ότι σαν όνομα project δώσαμε firstclass, και σαν project Location δώσαμε το δίσκο (c:) , τότε μετά τη μεταγλώττιση στο c: δημιουργήθηκε φάκελος με το όνομα του project δηλαδή firstclass. Το First.java βρίσκεται στο path → c:\firstclass\src. Το First.class βρίσκεται στο Path → c:\firstclass\build\classes.

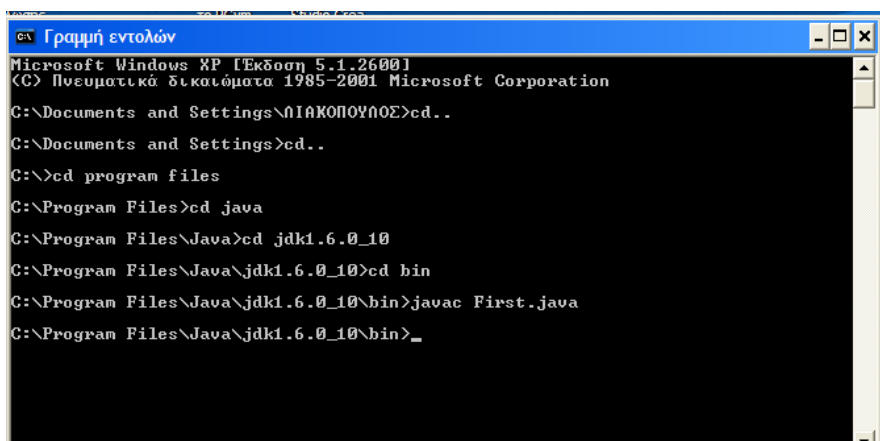
Τρόπος Β'

Μπορούμε να τρέξουμε το πρόγραμμα μας από τη γραμμή εντολών του MS-DOS. Ανοίγουμε ένα τέτοιο παράθυρο με το εξής τρόπο :

Έναρξη → Προγράμματα → βοηθήματα → Γραμμή Εντολών.
Το αρχείο μας First.java πρέπει να έχει σωθεί στο φάκελο bin του JDK, δηλαδή στο path : c:\program files\java\jdk1.6.0_10\bin. Η εντολή μεταγλώττισης ενός προγράμματος είναι :

Javac First.java

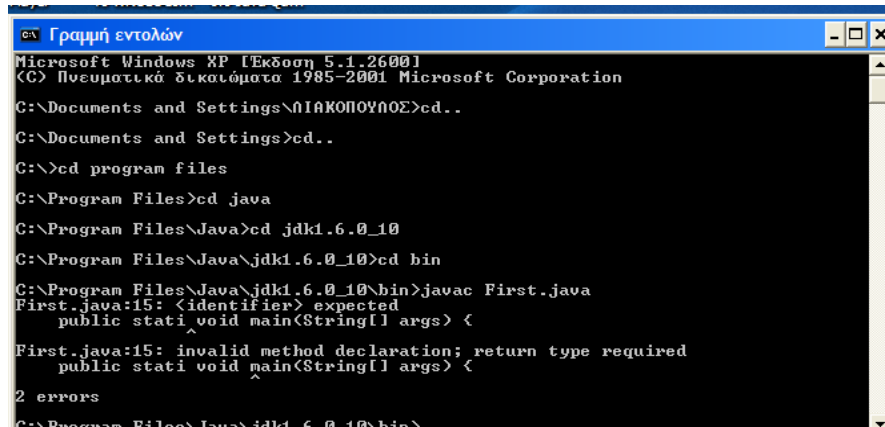
Αν δεν υπάρχουν συντακτικά λάθη τότε εμφανίζεται η παρακάτω οθόνη:



```

εν Γραμμή εντολών
Microsoft Windows XP [Έκδοση 5.1.2600]
(C) Πνευματικά δικαιώματα 1985-2001 Microsoft Corporation
C:\Documents and Settings\ΟΙΑΚΟΠΟΥΛΟΣ>cd..
C:\Documents and Settings>cd..
C:\>cd program files
C:\Program Files>cd java
C:\Program Files\Java>cd jdk1.6.0_10
C:\Program Files\Java\jdk1.6.0_10>cd bin
C:\Program Files\Java\jdk1.6.0_10\bin>javac First.java
C:\Program Files\Java\jdk1.6.0_10\bin>_
    
```

Αν υπάρχουν συντακτικά λάθη, τότε στη οθόνη της γραμμής εντολών εμφανίζονται τα λάθη όπως φαίνονται στη οθόνη που ακολουθεί :



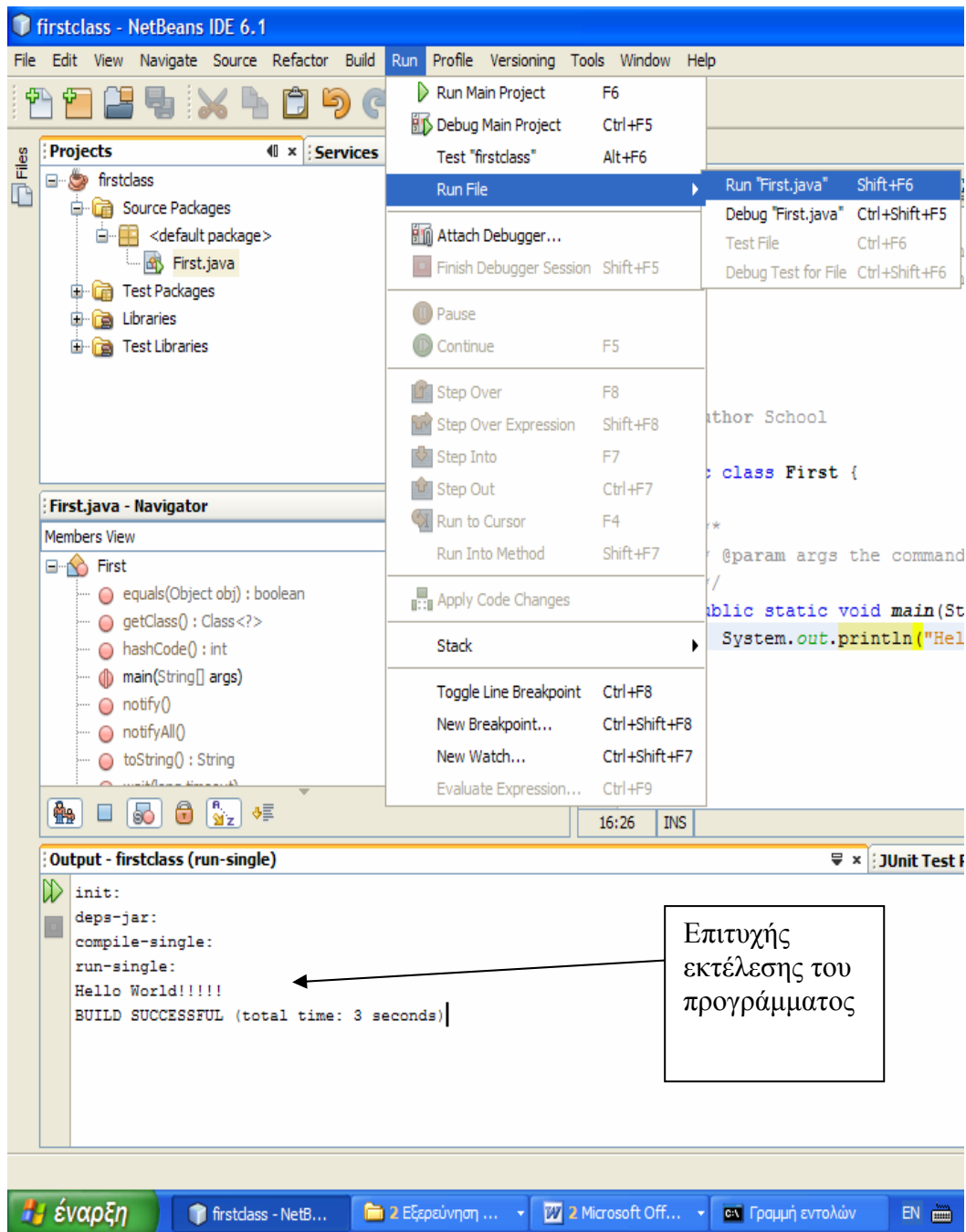
```
ca\ Γραμμή εντολών
Microsoft Windows XP [Έκδοση 5.1.2600]
(C) Πνευματικά δικαιώματα 1985-2001 Microsoft Corporation
C:\Documents and Settings\ΝΙΑΚΟΠΟΥΛΟΣ>cd..
C:\Documents and Settings>cd..
C:\>cd program files
C:\Program Files>cd java
C:\Program Files\Java>cd jdk1.6.0_10
C:\Program Files\Java\jdk1.6.0_10>cd bin
C:\Program Files\Java\jdk1.6.0_10\bin>javac First.java
First.java:15: <identifier> expected
    public stati void main(String[] args) {
           ^
First.java:15: invalid method declaration; return type required
    public stati void main(String[] args) {
           ^
2 errors
C:\Program Files\Java\jdk1.6.0_10\bin>
```

Εκτέλεση ενός προγράμματος

Τρόπος Α΄

Για να εκτελέσετε ένα πρόγραμμα από το NetBeans πρέπει :

Run → Run File → Run “First.java”. Παρακάτω φαίνεται η διαδικασία εκτέλεσης, και το αποτέλεσμα στη περιοχή του output.



Τρόπος Β'

Από το περιβάλλον του MS-DOS, για να εκτελέσουμε το First.class αρχείο, γράφουμε στο σημείο των εντολών την εξής εντολή :

Java First (Δεν χρειάζεται να γράψουμε τη προέκταση του αρχείου First.class). Το αποτέλεσμα φαίνεται στη οθόνη που ακολουθεί :

```

Microsoft Windows XP [Έκδοση 5.1.2600]
(C) Πνευματικά δικαιώματα 1985-2001 Microsoft Corporation

C:\Documents and Settings\ΝΙΑΚΟΠΟΥΛΟΣ>cd..
C:\Documents and Settings>cd..
C:\>cd program files
C:\Program Files>cd java
C:\Program Files\Java>cd jdk1.6.0_10
C:\Program Files\Java\jdk1.6.0_10>cd bin
C:\Program Files\Java\jdk1.6.0_10\bin>javac First.java
C:\Program Files\Java\jdk1.6.0_10\bin>java First
Hello World!!!!
C:\Program Files\Java\jdk1.6.0_10\bin>_

```

Εκτέλεση προγράμματος όπου απαιτείται εισαγωγή στοιχείων από το πληκτρολόγιο.

Ενδέχεται να χρειαστεί να εισάγουμε δεδομένα από το πληκτρολόγιο, κατά το χρόνο εκτέλεσης ενός προγράμματος. Αυτό γίνεται προσθέτοντας παραμέτρους στη εντολή του διερμηνευτή java, δηλαδή

Java όνομα προγράμματος όρισμα1 όρισμα2 όρισμα3

Η Java αποθηκεύει αυτά τα ορίσματα σε ένα πίνακα με το όνομα args. Το πρώτο όρισμα αντιστοιχεί στο στοιχείο args[0], το δεύτερο στο στοιχείο args[1] κλπ. Αν ο κώδικας του προγράμματος είναι ο παρακάτω, στη οθόνη που ακολουθεί, φαίνεται η διαδικασία μεταγλώττισης και εκτέλεσης.

```

public class second
{
    public static void main (String [] args)
    {
        int x1;
        double x2;

        x1= Integer.valueOf (args [0]).intValue ();
        x1 = x1* x1;
    }
}

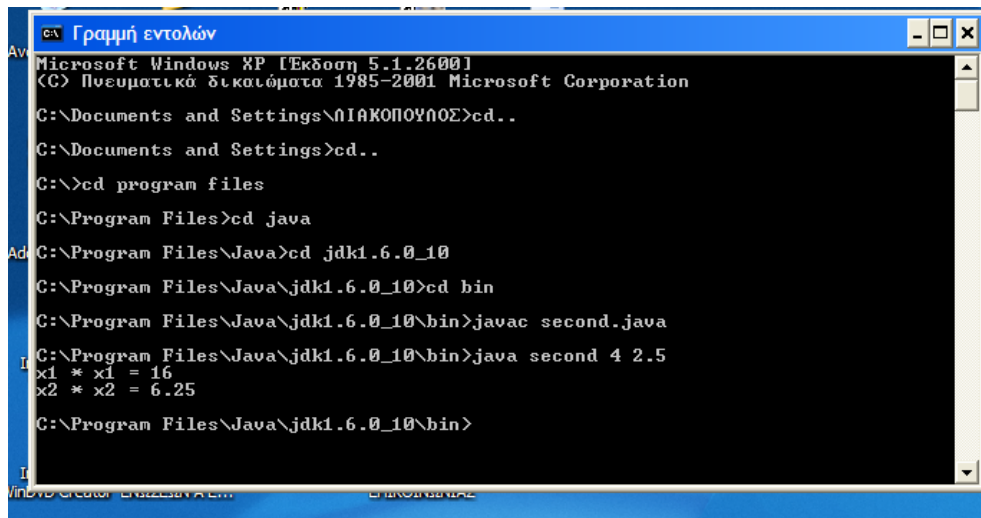
```

```
x2= Double.valueOf (args[1]).doubleValue ();  
x2=Math.pow(x2,2);
```

```
System.out.println ("x1 * x1 = " + x1);  
System.out.println ("x2 * x2 = " + x2);
```

```
}
```

```
}
```



```
Microsoft Windows XP [Έκδοση 5.1.2600]  
(C) Πνευματικά δικαιώματα 1985-2001 Microsoft Corporation  
  
C:\Documents and Settings\ΝΙΑΚΟΠΟΥΛΟΣ>cd..  
C:\Documents and Settings>cd..  
C:\>cd program files  
C:\Program Files>cd java  
C:\Program Files\Java>cd jdk1.6.0_10  
C:\Program Files\Java\jdk1.6.0_10>cd bin  
C:\Program Files\Java\jdk1.6.0_10\bin>javac second.java  
C:\Program Files\Java\jdk1.6.0_10\bin>java second 4 2.5  
x1 * x1 = 16  
x2 * x2 = 6.25  
C:\Program Files\Java\jdk1.6.0_10\bin>
```

ΕΝΟΤΗΤΑ 1

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

- 1) Να μπορέσουν οι μαθητές να διακρίνουν ποια θα είναι τα στοιχεία εισόδου, ποια η επεξεργασία και ποια η έξοδος ενός προγράμματος.
- 2) Πως θα δηλώσουν μία μεταβλητή και τι τύπο θα ορίσουν για κάθε μεταβλητή μέσα στο πρόγραμμα τους.
- 3) Να καταλάβουν, ότι για να δηλώσουν μία μεταβλητή, πρέπει πρώτα να υπάρχει ο τύπος της μεταβλητής και μετά το όνομα της.
- 4) Να συνειδητοποιήσουν, τι τύπο θα προσδιορίσουν για κάθε μεταβλητή, γιατί αφενός έχει σημασία η ταχύτητα εκτέλεσης του προγράμματος και αφετέρου να μη χάνονται σημαντικά ψηφία (significant digits) κατά τη επεξεργασία. Πχ αν θα δηλώσουν int ή float μία μεταβλητή.
- 5) Να ορίζουν μία μεταβλητή σύμφωνα με τους κανόνες που απαιτεί η Java και κυρίως να μη χρησιμοποιούν δεσμευμένες λέξεις.
- 6) Να καταλάβουν πότε θα χρησιμοποιούν τους τύπους δεδομένων byte, short, int, long, float και double.
- 7) Να καταλάβουν τη διαφορά χαρακτήρα και αλφαριθμητικού, καθώς και τη χρήση της λογικής μεταβλητής boolean.

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1'

Ποια από τις παρακάτω είναι μία έγκυρη τιμή για μια μεταβλητή boolean; A) "false" B) false C) 10 D) int number;

Άσκηση 2'

Ποια από τις παρακάτω δεν είναι μια σύμβαση ονοματοδοσίας μεταβλητών στη Java;

- A) Μετά από τη πρώτη λέξη στο όνομα της μεταβλητής, κάθε επόμενη λέξη αρχίζει με κεφαλαίο γράμμα.
- B) Το πρώτο γράμμα του ονόματος της μεταβλητής είναι πεζό.
- C) Όλα τα γράμματα είναι κεφαλαία.

Άσκηση 3'

Ποιος από τους παρακάτω τύπους δεδομένων περιέχει τιμές από -32768 έως 32768; A) char B) short C) byte D) float.

Άσκηση 4'

Ποια από τις παρακάτω μεταβλητές δεν είναι έγκυρη;

- A) diaMeter B) _var C) \$local D) 7var

Άσκηση 5'

Μετά από τις δηλώσεις των παρακάτω μεταβλητών, ποια θα είναι η τιμή της μεταβλητής num1;

Int num, num1, num2 = 100;

- A) 0 B) 49 C) 50 D) δεν είναι ορισμένη.

Άσκηση 6'

Ποια η διαφορά των τύπων μεταβλητών byte και int; Ποια η διαφορά ενός χαρακτήρα και ενός αλφαριθμητικού; Για ποιο σκοπό χρησιμοποιούμε τη boolean δήλωση μεταβλητής;

ΕΝΟΤΗΤΑ 2

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

1. Δηλώνετε τις μεταβλητές του προγράμματος και το κάθε τύπο.
2. Να χρησιμοποιείτε τους τελεστές αριθμητικών πράξεων.
3. Να χρησιμοποιείτε τους μοναδιαίους τελεστές αύξησης και μείωσης.
4. Να καταχωρείτε τιμή σε μεταβλητή.
5. Να χρησιμοποιείτε τους τελεστές σύγκρισης καθώς και τους λογικούς τελεστές.
6. Να καταλάβετε τη διαφορά μεταξύ ενός χαρακτήρα και ενός αλφαριθμητικού.
7. Να κάνετε μετατροπή από ένα τύπο δεδομένων σε άλλο.
8. Να μετατρέπετε αλφαριθμητικά σε αριθμούς.
9. Να χρησιμοποιείτε τις συναρτήσεις της κλάσης Math.
10. Να εισάγετε δεδομένα από το πληκτρολόγιο.
11. Να κατασκευάζετε τα δικά σας προγράμματα.

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1'

Ποια θα είναι η τιμή της μεταβλητής z μετά τη εκτέλεση του παρακάτω κώδικα; Int z=100;

 ++z;

- a) 100 b) 99 c) 101 d) αόριστη

Άσκηση 2'

Ποιο το αποτέλεσμα της πράξης : 14%5;

- a) 4 b) 2 c) 14 d) 0

Άσκηση 3'

Ποιο το αποτέλεσμα των παρακάτω πράξεων ;

- a) (10 == 6+4) && (5<7)
b) (5<7)|| (10 == 11)|| (3!=3)
c) (4!=5) && ((5 == 3+2)|| (5<5+2));

Άσκηση 4'

Να δημιουργηθεί πρόγραμμα, που να δίνονται τιμές σε δύο ακέραιες μεταβλητές και στη συνέχεια να υπολογίζεται και να τυπώνεται το άθροισμα, η διαφορά, το γινόμενο, το ακέραιο πηλίκο και το πηλίκο. Για όλα τα αποτελέσματα να χρησιμοποιηθούν ακέραιες μεταβλητές, ενώ για το πηλίκο μεταβλητές τύπου double.

Άσκηση 5'

Να υπολογισθεί το εμβαδόν και το μήκος της περιφέρειας κύκλου, όταν δίνεται η ακτίνα του.

Άσκηση 6'

Να γραφεί πρόγραμμα το οποίο να δέχεται από το πληκτρολόγιο ένα ποσό μάζας m και να βρίσκει το ισοδύναμο ποσό ενέργειας E , με βάση τον τύπο της φυσικής $E = m c^2$ όπου c η ταχύτητα του φωτός ($c = 300.000 \text{ km/sec}$).

Άσκηση 7'

Ο ολική αντίσταση $R_{ολ}$ δύο αντιστάσεων $R1$ & $R2$ που συνδέονται σε σειρά, είναι $R_{ολ} = R1 + R2$. Να γράψετε πρόγραμμα το οποίο να δέχεται σαν εισόδους τις τιμές των $R1$ & $R2$ και να βρίσκει και να τυπώνει την συνολική αντίσταση $R_{ολ}$.

Άσκηση 8'

Να φτιαχτεί πρόγραμμα που να υπολογίζει τις πραγματικές ρίζες μίας εξίσωσης Δευτέρου Βαθμού της μορφής : $ax^2 + bx + c = 0$

Άσκηση 9'

Να δημιουργηθεί πρόγραμμα που να δίνονται δύο αριθμοί από το πληκτρολόγιο και στην συνέχεια ο πρώτος να μετατρέπεται σε δεδομένο τύπου int, ενώ ο δεύτερος σε double. Στην συνέχεια το πρόγραμμα να βρίσκει και να τυπώνει τα τετράγωνά τους

ΕΝΟΤΗΤΑ 3

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

1. καταλάβετε τη χρήση της δομής if ... else.
2. Χρησιμοποιείτε τα ένθετα (nested) if.
3. Να χρησιμοποιείτε το τελεστή ? :
4. Χρησιμοποιείτε τη εντολή switch.
5. Να εξασκηθείτε επαρκώς στη υλοποίηση προγραμμάτων.

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1΄

Στο παρακάτω κομμάτι προγράμματος η μεταβλητή selection έχει τη τιμή 10. θα εκτελεστεί η δεύτερη γραμμή του προγράμματος;

```
If (selection == 3)  
    Number = selection;
```

Άσκηση 2΄

Να ερμηνεύσετε τη λειτουργία του παρακάτω προγράμματος :

```
Switch (x)  
{  
    Case 1 :  
        Case 2 :  
            Y = 1;  
            Break;  
        Case 3 :  
            Y = 2;  
            Break;  
        Default :  
            Y = 5;  
}
```

Άσκηση 3΄

Να δημιουργηθεί πρόγραμμα που να διαβάζει τρεις διαφορετικούς ακεραίους x,y,z. Να τοποθετηθούν σε φθίνουσα σειρά.

Άσκηση 4'

Να γραφεί πρόγραμμα που να βρίσκει ένα τυχαίο αριθμό από το 1 μέχρι το 7 και ανάλογα να τυπώνει τη αντίστοιχη μέρα της εβδομάδας.

Άσκηση 5'

Δίνεται ότι:

```
Public class cases {
    Public static void main (String [] args) {
        Float x = 9;
        Float y = 5;
        Int z = (int) (x / y);
        Switch (z) {
            Case 1:
                X = x + 2;
            Case 2:
                X = x + 3;
            Default:
                X = x + 1;
        }
        System.out.println ("value of x : " + x );
    }
}
```

Ποια θα είναι η τιμή της x όταν εκτελεστεί το παραπάνω κομμάτι προγράμματος; Α) 9.0 Β) 11.0 Γ) 15.0 Δ) δεν θα μεταγλωττιστεί.

Άσκηση 6'

Να γραφεί πρόγραμμα το οποίο να διαβάζει από το πληκτρολόγιο ένα γράμμα. Αν ο χρήστης πάτησε το γράμμα 'E' τότε το πρόγραμμα να υπολογίζει το εμβαδόν ενός κύκλου. Αν πάτησε οποιοδήποτε άλλο πλήκτρο τότε το πρόγραμμα να υπολογίζει τη περίμετρο του κύκλου.

Άσκηση 7'

Να διαβαστούν δύο αριθμοί και σε περίπτωση που ο πρώτος αριθμός, είναι μικρότερος του δεύτερου, να υπολογιστεί και να εκτυπωθεί το άθροισμά τους. Διαφορετικά να υπολογιστεί και να εκτυπωθεί, το γινόμενο τους.

Άσκηση 8'

Να εισαχθεί ένας ακέραιος που αντιστοιχεί σε μία ηλικία και να βρεθεί σε ποια όρια, εντάσσεται η δεδομένη ηλικία, εμφανίζοντας σχετικό μήνυμα.

Άσκηση 9'

Να διαβάζονται δύο αριθμοί που αντιστοιχούν στο ύψος και στο βάρος ενός άντρα. Να εκτυπώνεται ότι ο άντρας είναι «ελαφρύς» αν το βάρος του είναι κάτω από 80 κιλά, ή να εκτυπώνεται «βαρύς» στην αντίθετη περίπτωση. Επίσης να εκτυπώνεται «κοντός» αν το ύψος του είναι κάτω από 1,70, αλλιώς να εκτυπώνεται «ψηλός».

Άσκηση 10'

Στο λογαριασμό της ΔΕΗ η κατανάλωση υπολογίζεται από την διαφορά μεταξύ της προηγούμενης και της τελευταίας ένδειξης. Η αξία του λογαριασμού προκύπτει ως εξής :

- Υπάρχει πάγια χρέωση 15 ευρώ
- Για κατανάλωση μέχρι 800 KWH υπάρχει χρέωση 1,20 ευρώ/ kwh
- Για κατανάλωση από 801 kwh έως 1.500 kwh, υπάρχει χρέωση 1,70 euro/kwh.
- Για κατανάλωση από 1.501 kwh έως 2.000 kwh, υπάρχει χρέωση 2,5 euro/kwh.
- Για κατανάλωση πάνω από 2.000 kwh, υπάρχει χρέωση 3,00 euro/kwh.

Να γίνει πρόγραμμα που να ζητά την τελευταία και την προηγούμενη ένδειξη και να υπολογίζει την αξία του λογαριασμού.

ΕΝΟΤΗΤΑ 4

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

1. Να καταλάβετε το σκοπό ύπαρξης των βρόχων.
2. Να δείτε πως δουλεύει ο βρόχος for, while, και do – while.
3. Να συνειδητοποιήσετε πότε θα χρησιμοποιείτε το κάθε βρόχο.
4. Να μπορείτε να μετατρέπετε τη δομή ενός προγράμματος από τον ένα βρόχο στο άλλο βρόχο.
5. Να καταλάβετε τη βασική διαφορά ανάμεσα στους βρόχους while και do-while.
6. Να καταλάβετε γιατί χρησιμοποιούμε το ατέρμονο βρόχο.
7. Να μπορείτε να χρησιμοποιείτε τους ένθετους βρόχους.
8. Να χρησιμοποιείτε το continue και το continue με label.
9. Να χρησιμοποιείτε το break και το break με label.

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1'

Πότε χρησιμοποιούμε το βρόχο for και πότε while ή do-while; Ποια η διαφορά χρήσης του while και του do-while; Πότε ένας βρόχος while παύει να εκτελείται; Πότε συμβαίνει το ίδιο σε ένα βρόχο for;

Άσκηση 2'

Γράψτε ένα βρόχο while ο οποίος να εκτελείται 10 φορές. Πώς μπορούμε να γράψουμε τον ίδιο βρόχο χρησιμοποιώντας το for;

Άσκηση 3'

Πόσες φορές θα εκτελεστεί ο παρακάτω βρόχος :

```
For (int i=3; i< 15; i += 3)
{
    ++count;
}
```

Άσκηση 4'

Τι κάνει το παρακάτω κομμάτι προγράμματος :

```
For (int i=1; i<10; i++)  
{  
    For (int j=1; j<10; j++)  
        System.out.println ("*");  
    System.out.println ();  
}
```

Άσκηση 5'

Να γραφεί πρόγραμμα το οποίο να τυπώνει πίνακα αντιστοιχιών θερμοκρασίας Κελσίου και Φαρενάιτ από 0 μέχρι 50 Φαρενάιτ. Ο τύπος μετατροπής είναι : $C = 5 / 9 (F - 32)$.

Άσκηση 6'

Να δημιουργηθεί πρόγραμμα το οποίο να παράγει ένα τυχαίο αριθμό μεταξύ του 1 και του 100 και στη συνέχεια να εξετάζει αν ο αριθμός αυτός είναι περιττός ή άρτιος. Σε κάθε περίπτωση να τυπώνει και το αντίστοιχο μήνυμα.

Άσκηση 7'

Να δημιουργηθεί πρόγραμμα που να διαβάζει τρεις διαφορετικούς ακεραίους x,y,z. Να τοποθετηθούν σε φθίνουσα σειρά.

Άσκηση 8'

Να γίνει πρόγραμμα στο οποίο να δίνεται ένας ακέραιος αριθμός και στη συνέχεια να υπολογίζεται το άθροισμα όλων των αριθμών. Να γίνει η συγκεκριμένη άσκηση και με τους τρεις βρόχους.

Άσκηση 9'

Να δημιουργηθεί πρόγραμμα το οποίο να δίνει σε μία μεταβλητή y τη τιμή τύπου float 20.0. Στη συνέχεια η μεταβλητή αυτή να υποδιπλασιάζεται μέχρι να γίνει μικρότερη από τη τιμή 0.01.

Άσκηση 10'

Χρησιμοποιώντας ένθετους βρόχους, να γίνει πρόγραμμα που να κατασκευάζει τη προπαίδεια του 1,2,3,4,5,6,7,8,9,10.

Άσκηση 11'

Να γραφεί πρόγραμμα που να βρίσκει ένα τυχαίο αριθμό από το 1 μέχρι το 7 και ανάλογα να τυπώνει τη αντίστοιχη μέρα της εβδομάδας

Άσκηση 12'

Να γραφεί πρόγραμμα που να υπολογίζει το άθροισμα των περιττών και των άρτιων αριθμών που είναι μικρότεροι από το 100.

Άσκηση 13'

Να γραφεί πρόγραμμα το οποίο να παράγει 50 τυχαίους αριθμούς από το 1 μέχρι το 100 και να υπολογίζει το άθροισμα και το μέσο όρος τους.

Άσκηση 14'

Στο λογαριασμό της ΔΕΗ η κατανάλωση υπολογίζεται από την διαφορά μεταξύ της προηγούμενης και της τελευταίας ένδειξης. Η αξία του λογαριασμού προκύπτει ως εξής :

- Υπάρχει πάγια χρέωση 15 ευρώ
- Για κατανάλωση μέχρι 800 KWH υπάρχει χρέωση 1,20 ευρώ/ kwh
- Για κατανάλωση από 801 kwh έως 1.500 kwh, υπάρχει χρέωση 1,70 euro/kwh.
- Για κατανάλωση από 1.501 kwh έως 2.000 kwh, υπάρχει χρέωση 2,5 euro/kwh.
- Για κατανάλωση πάνω από 2.000 kwh, υπάρχει χρέωση 3,00 euro/kwh.

Να γίνει πρόγραμμα που να ζητά την τελευταία και την προηγούμενη ένδειξη καθώς και το αριθμό μετρητή του ρολογιού της ΔΕΗ. Να υπολογίζει την αξία του λογαριασμού για N συνδρομητές και το πρόγραμμα να σταματά τη εκτέλεση του, όταν σαν αριθμό μετρητή πατήσουμε το 0.

ΕΝΟΤΗΤΑ 5

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

- 1) Να ορίσετε πίνακα μιας διάστασης, να καταχωρήσετε τιμές στο πίνακα και να τον επεξεργαστείτε.
- 2) Να καταλάβετε το λόγο ύπαρξης των πινάκων πολλαπλών διαστάσεων και την επεξεργασία τους.
- 3) Να κατανοήσετε τις δύο μορφές ταξινόμησης.

Ασκήσεις με λυμένα παραδείγματα για εμπέδωση της ύλης

Άσκηση 1'

Το παράδειγμα που ακολουθεί είναι ένα παράδειγμα ενός βρόχου while που αντιγράφει τα στοιχεία ενός πίνακα ακεραίων σε ένα πίνακα αριθμών Float, αλλάζοντας τον τύπο κάθε στοιχείου σε Float.

```
class Copying {
    public static void main (String [] arguments) {
        int [] array1 = { 7, 4, 8, 1, 4, 1, 4 };
        float [] array2 = new float [array1.length];

        System.out.print ("array1: [ ");
        for (int i = 0; i < array1.length; i++) {
            System.out.print (array1 [i] + " ");
        }
        System.out.println ("]");

        System.out.print ("array2: [ ");
        int count = 0;
        while (count < array1.length && array1 [count] != 1) {
            array2 [count] = (float) array1 [count];
            System.out.print (array2 [count++] + " ");
        }
        System.out.println ("]");
    }
}
```

Άσκηση 2'

Το παρακάτω πρόγραμμα δημιουργεί, αρχικοποιεί, τροποποιεί και εξετάζει τα μέρη ενός πίνακα.

```
class half {
    public static void main (String [] arguments) {
        int [] denver = {15000006, 18810000, 20752110};
        int [] philadelphia = new int [denver.length];
        int [] total = new int [denver.length];
        int average;

        philadelphia [0] = 15020000;
        philadelphia [1] = 18708000;
        philadelphia [2] = 21348000;

        total [0] = denver [0] + philadelphia [0];
        total [1] = denver [1] + philadelphia [1];
        total [2] = denver [2] + philadelphia [2];
        average = (total [0] + total [1] + total [2]) / 3;

        System.out.println ("1993 production: " + total [0]);
        System.out.println ("1994 production: " + total [1]);
        System.out.println ("1995 production: " + total [2]);
        System.out.println ("Average production: "+ average);
    }
}
```

Άσκηση 3'

Η παραπάνω άσκηση 2, τροποποιημένη χρησιμοποιώντας βρόχους for, για να εξαλειφθεί ο επαναλαμβανόμενος κώδικας.

```
class half {
    public static void main (String [] arguments) {
        int [] denver = {15000006, 18810000, 20752110};
        int [] philadelphia = {15020000, 18708000, 21348000};
        int [] total = new int [denver.length];
        int sum = 0;

        for (int i = 0; i < denver.length; i++) {
            total [i] = denver [i] + philadelphia [i];
            System.out.println ((i + 1993) + " production: "
                + total [i]);
        }
    }
}
```

```
        sum += total [i];  
    }  
  
    System.out.println ("Average production: "  
        + (sum / denver.length));  
    }  
}
```

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1΄

Με ποιο τρόπο δηλώνουμε ένα πίνακα 30 θέσεων; Πώς θα αναφερθούμε σε συγκεκριμένο στοιχεί του πίνακα; Ποια η μεγαλύτερη τιμή του δείκτη; Τι θα συμβεί αν προσπαθήσουμε να προσπελάσουμε ένα στοιχείο έξω από τα όρια του πίνακα;

Άσκηση 2΄

Δηλώστε ένα πίνακα δύο διαστάσεων με 5 γραμμές και 4 στήλες. Γράψτε το διπλό for που απαιτείται για να ορίσετε τον πίνακα, να αλλάξετε τη τιμή κάποιου στοιχείου του και να τον εκτυπώσετε.

Άσκηση 3΄

Να δημιουργηθεί ένας πίνακας δέκα θέσεων και να καταχωρηθούν σε αυτόν τυχαίες τιμές. Να γίνει πρόγραμμα που να σχηματίζει το πίνακα και να υπολογισθούν :

- A) το άθροισμα όλων των στοιχείων του πίνακα.
- B) το μεγαλύτερο στοιχείο του πίνακα και σε ποια θέση εμφανίζεται.
- Γ) αν το μεγαλύτερο στοιχείο του πίνακα υπάρχει σε περισσότερες από μία θέσεις να αντικατασταθεί η τιμή αυτή με το μηδέν.
- Δ) Να ταξινομηθεί ο τροποποιημένος πίνακας.

Άσκηση 4΄

Να δημιουργηθεί ένας πίνακας δύο διαστάσεων με τρεις γραμμές και τρεις στήλες. Να γίνει πρόγραμμα που να σχηματίζει το πίνακα αυτό και να υπολογίζει :

- A) τον αντίστροφο του σε ένα άλλο πίνακα.
- B) να δημιουργηθούν δύο άλλοι πίνακες που θα προκύψουν από τη πρόσθεση και τη αφαίρεση του βασικού και του αντίστροφου.

- Γ) για το βασικό πίνακα να βρίσκει το άθροισμα όλων των στοιχείων του.
- Δ) για το βασικό πίνακα να βρίσκει το άθροισμα της κάθε στήλης.
- Ε) για το βασικό πίνακα να βρίσκει το μεγαλύτερο της κάθε γραμμής.
- Ζ) για το αντίστροφο πίνακα να βρίσκει το άθροισμα της κυρίας διαγώνιου.

ΕΝΟΤΗΤΑ 6

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

1. Να καταλάβετε τι είναι μέθοδος (function).
1. Να καταλάβετε πως δηλώνετε μία μέθοδος.
2. Να καταλάβετε πώς θα επιστρέψει τιμή μία μέθοδος.
3. Να δείτε το τρόπο πως θα καλέσετε μία μέθοδο.
4. Να καταλάβετε τη χρησιμότητα μιας αναδρομικής μεθόδου.

Ασκήσεις με λυμένα παραδείγματα για εμπέδωση της ύλης

Άσκηση 1'

Το παρακάτω πρόγραμμα παίρνει δύο ορίσματα, ένα μήνα και ένα έτος, και εμφανίζει τον αριθμό των ημερών μέσα στο μήνα.

```
class Day {
    public static void main (String [] arguments) {
        int yearIn = 2002;
        int monthIn = 12;
        if (arguments.length > 0)
            monthIn = Integer.parseInt (arguments [0]);
        if (arguments.length > 1)
            yearIn = Integer.parseInt (arguments [1]);
        System.out.println (monthIn + "/" + yearIn + " has "
            + countDays (monthIn, yearIn) + " days.");
    }

    static int countDays (int month, int year) {
        int count = -1;
        switch (month) {
            case 1:
            case 3:
            case 5:
            case 7:
            case 8:
            case 10:
```

```

    case 12:
        count = 31;
        break;
    case 4:
    case 6:
    case 9:
    case 11:
        count = 30;
        break;
    case 2:
        if (year % 4 == 0)
            count = 29;
        else
            count = 28;
        if ((year % 100 == 0) & (year % 400 != 0))
            count = 28;
    }
    return count;
}
}

```

Άσκηση 2'

Δημιουργήστε μία εφαρμογή που εμφανίζει κάθε ημερομηνία μέσα σε ένα δεδομένο έτος σε μία λίστα από την 1 Ιανουαρίου μέχρι τη 31 Δεκεμβρίου.

```

class Year {
    public static void main (String [] arguments) {
        int year = 2000;
        if (arguments.length > 0)
            year = Integer.parseInt (arguments [0]);
        for (int month = 1; month < 13; month++)
            for (int day = 1; day <= countDays (month, year); day++)
                System.out.println (month + "/" + day + "/" + year);
    }

    static int countDays (int month, int year) {
        int count = -1;
        switch (month) {
            case 1:
            case 3:
            case 5:

```

```

case 7:
case 8:
case 10:
case 12:
    count = 31;
    break;
case 4:
case 6:
case 9:
case 11:
    count = 30;
    break;
case 2:
    if (year % 4 == 0)
        count = 29;
    else
        count = 28;
    if ((year % 100 == 0) & (year % 400 != 0))
        count = 28;
}
return count;
}
}

```

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1'

Να δημιουργηθεί πρόγραμμα και μέσα σε αυτό να δημιουργηθεί η μέθοδος tetragon(), η οποία να βρίσκει το τετράγωνο ενός ακεραίου αριθμού. Το κυρίως πρόγραμμα να καλεί είκοσι φορές τη μέθοδο tetragon() για να βρίσκει και να τυπώνει τα τετράγωνα των αριθμών από 0 έως 19

Άσκηση 2'

Να δημιουργηθεί πίνακας 10 στοιχείων με τυχαίους αριθμούς από το 0 μέχρι το 49. Να γίνει πρόγραμμα με συναρτήσεις (μεθόδους) που να :

A) η συνάρτηση να επιστρέφει στο κυρίως πρόγραμμα το άθροισμα όλων των στοιχείων του πίνακα.

B) άλλη συνάρτηση να επιστρέφει στο κυρίως πρόγραμμα το μεγαλύτερο στοιχείο του πίνακα.

ΕΝΟΤΗΤΑ 7

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

1. Μία μεταβλητή αλφαριθμητικού τύπου είναι ένα αντικείμενο της κλάσης String. Με τον όρο *αλφαριθμητικά* (*string*) εννοούμε μία σειρά από χαρακτήρες πχ ένα όνομα
2. Για *συνένωση* δύο ή περισσότερων αλφαριθμητικών χρησιμοποιούμε τον τελεστή +. Για *έλεγχο ισότητας* δύο αλφαριθμητικών χρησιμοποιείται η μέθοδος *equals*. Η κλάση String περιέχει μεθόδους με τις οποίες μπορούμε να *μετατρέψουμε τα κεφαλαία γράμματα σε πεζά και αντίστροφα*. Αν θέλουμε να μη λαμβάνεται κατά τη σύγκριση υπ' όψη το αν οι χαρακτήρες είναι κεφαλαίο ή πεζοί, τότε χρησιμοποιούμε τη μέθοδο *equalsIgnoreCase()* αντί για τη *equals*.
3. Με τον όρο *υπααλφαριθμητικό* (*substring*) εννοούμε ένα αλφαριθμητικό, το οποίο βρίσκεται μέσα σε ένα αλφαριθμητικό μεγαλύτερο ή ίσο από αυτό. Για τη εξαγωγή ενός αλφαριθμητικού από ένα άλλο χρησιμοποιούμε τη μέθοδο *substring()*. Η σύνταξη της είναι : `String substring (int αρχή , int τέλος)`
3. Πολλές φορές χρειαζόμαστε να πάρουμε μεμονωμένους χαρακτήρες από ένα αλφαριθμητικό. Χρησιμοποιούμε μια μεταβλητή τύπου `int` για να προσδιορίσουμε τη θέση του χαρακτήρα σε συνδυασμό με τη μέθοδο *charAt()*
3. Η μέθοδος *length()*, μία χρήσιμη μέθοδος, για να μπορέσουμε να βρούμε το μήκος του αλφαριθμητικού.
3. Η μέθοδος *replace()* χρησιμοποιείται για να αντικατασταθούν κάποιοι χαρακτήρες ενός αλφαριθμητικού με άλλους. Η μέθοδος *replace()* δέχεται δύο παραμέτρους. Ο πρώτος είναι ο χαρακτήρας που πρόκειται να αντικατασταθεί, και ο δεύτερος ο χαρακτήρας που θα πάρει τη θέση του.
3. Οι μέθοδοι *indexOf()* και *lastIndexOf()* είναι δύο μέθοδοι που χρησιμοποιούνται για τον εντοπισμό χαρακτήρων μέσα σε ένα αλφαριθμητικό.

3. Στη Java έχουμε τη δυνατότητα να μετατρέψουμε διάφορους τύπους δεδομένων σε αλφαριθμητικά με τη μέθοδο *valueOf()*.
3. Η μέθοδος *parseInt()* διαβάζει το περιεχόμενο του αλφαριθμητικού, το μετατρέπει σε ακέραιο, και στη συνέχεια το καταχωρεί σε κάποια μεταβλητή.

Ασκήσεις με λυμένα παραδείγματα για εμπέδωση της ύλης

Παράδειγμα 1'

Παράδειγμα ελέγχου ισότητας δύο αλφαριθμητικών

```
public class b7
{
    public static void main (String [] args)
    {
        String string1 = "George";
        String string2 = "george";
        String string3 = "George";

        System.out.println (string1 + "\n" + string2 + "\n" + string3);

        if (string1.equals (string3))
            System.out.println ("string1 is equal to string3");
        else
            System.out.println ("string1 is not equal to string3");
        if (string2.equals (string3))
            System.out.println ("string2 is equal to string3");
        else
            System.out.println ("string2 is not equal to string3");
    }
}
```

Παράδειγμα 2'

Παράδειγμα ορισμού του υποαλφαριθμητικού (substring)

```
public class b9
{
    public static void main (String [] args)
    {
```

```
String str1 = "Athens is the capital of Greece";
System.out.println (str1);
String str2 = str1.substring (14,21);
System.out.println (str2);
String str3 = str1.substring (25,31);
System.out.println (str3);
}
}
```

Παράδειγμα 3'

Εύρεση του μήκους ενός αλφαριθμητικού και προσπέλαση μεμονωμένων χαρακτήρων.

```
public class b10
{
    public static void main (String [] args)
    {
        String str1 = "Athens is the capital of Greece";
        System.out.println ("the character at index 4 = " +
            str1.charAt (4));
        System.out.println ("the string contains = " +
            str1.length() + " characters");
    }
}
```

Παράδειγμα 4'

Παράδειγμα εντοπισμού χαρακτήρων μέσε σε ένα αλφαριθμητικό.

```
public class b12
{
    public static void main (String [] args)
    {
        String str1 = "Athens is the Capital of Greece.";
        System.out.println (str1);
        int i = 0;
        int j = 0;
        int k = 0;
        i = str1.indexOf ('e');
        System.out.println ("the first index of 'e' = " + i);
        j = str1.indexOf ('e', i+1);
        System.out.println ("the next index of 'e' = " + j);
        k = str1.lastIndexOf ('e');
    }
}
```

```

System.out.println ("the last index of 'e' = " + k);
System.out.println (str1.substring (i));
}
}

```

Παράδειγμα 5'

Παράδειγμα χρήσης της substring().

```

class Birthday {
    public static void main (String [] arguments) {
        String birthday = "08/23/2002";
        String month = birthday.substring (0, 2);
        String day = birthday.substring (3, 5);
        String year = birthday.substring (6, 10);
        System.out.println ("Birthday: " + birthday);
        System.out.println ("Month: " + month);
        System.out.println ("Day: " + day);
        System.out.println ("Year: " + year);    }
}

```

Παράδειγμα 6'

Παράδειγμα χρήσης των μεθόδων length(), charAt(), indexOf()

```

class CheckString {

    public static void main (String [] arguments) {
        String str = "Nobody ever went broke by buying IBM";
        System.out.println ("The string is: " + str);
        System.out.println ("Length of this string: "
            + str.length ());
        System.out.println ("The character at position 5: "
            + str.charAt (5));
        System.out.println ("The substring from 26 to 32: "
            + str.substring (26, 32));
        System.out.println ("The index of the character v: "
            + str.indexOf ('v'));
        System.out.println ("The index of the beginning of the "
            + "substring \"IBM\": " + str.indexOf ("IBM"));
        System.out.println ("The string in upper case: "
            + str.toUpperCase ());
    }
}

```

Παράδειγμα 7'

Δημιουργείστε μια κλάση με μεταβλητές τα height, weight, depth, κάνοντας κάθε μεταβλητή ακέραιο. Δημιουργείστε μία εφαρμογή που χρησιμοποιεί τη παραπάνω κλάση, θέτει κάθε μια από τις τιμές σε ένα αντικείμενο και εμφανίζει τις τιμές.

```
class Sample {  
    int height;  
    int weight;  
    int depth;  
}
```

```
class UseSample {  
    public static void main (String [] arguments) {  
        Sample thing = new Sample ();  
        thing.height = 72;  
        thing.weight = 1000;  
        thing.depth = 420;  
        System.out.println ("Height: " + thing.height);  
        System.out.println ("Weight: " + thing.weight);  
        System.out.println ("Depth: " + thing.depth);  
    }  
}
```

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1'

Σε ένα string, να μετρηθεί ο αριθμός των λέξεων «the» που υπάρχουν μέσα σε μια δεδομένη φράση.

Άσκηση 2'

Γράψτε ένα πρόγραμμα το οποίο να δέχεται τη ημερομηνία σε μορφή «ηη/μμ/εε» και να τη μετατρέπει σε μορφή «ηη/μμ/20εε» δηλαδή την ημερομηνία 15/07/08 να την μετατρέψει σε 15/07/2008.

Άσκηση 3'

Δημιουργήστε ένα πρόγραμμα που δέχεται λέξεις για τους δέκα πρώτους αριθμούς (ένα έως δέκα) και να τις μετατρέπει σε ένα ακέραιο long. Χρησιμοποιείτε μια πρόταση switch για τη μετατροπή και ορίσματα γραμμής εντολών για τις λέξεις.

Άσκηση 4'

Δημιουργήστε ένα πρόγραμμα που να ελέγχει αν δύο αντικείμενα τύπου String έχουν τις ίδιες τιμές.

Άσκηση 5'

Γράψτε πρόγραμμα που να μετατρέπει μία ημερομηνία γέννησης της μορφής MM/HH/EEEE σε τρία διαφορετικά strings.

ΕΝΟΤΗΤΑ 8

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

1. Σε ένα αντικείμενο της κλάσης String δεν μπορούμε να αλλάξουμε ένα αλφαριθμητικό μέσα στο ίδιο το αντικείμενο. Η κλάση StringBuffer χρησιμοποιείται για αλφαριθμητικά που μεταβάλλονται.
2. Βασικές μέθοδοι της κλάσης StringBuffer είναι :

Length () → μας δίνει το χρησιμοποιούμενο μήκος
Capacity () → μας δίνει τη συνολική χωρητικότητα του String.
Append () → προσθέτουμε χαρακτήρες στο αντικείμενο
3. Οι μέθοδοι *setCharAt()* και *insert()* χρησιμοποιούνται για να κάνουμε προσθήκες στο αλφαριθμητικό περιεχόμενο ενός αντικειμένου τύπου StringBuffer.
4. Ενδέχεται ένα αντικείμενο της κλάσης StringBuffer να θέλουμε να μετατραπεί σε αλφαριθμητικό, για αυτό χρησιμοποιούμε τη μέθοδο *toString ()*.
5. Η μέθοδος *reverse()* της κλάσης StringBuffer, χρησιμοποιείται για να γίνει αντιστροφή ενός αλφαριθμητικού τύπου StringBuffer.

Ασκήσεις με λυμένα παραδείγματα για εμπέδωση της ύλης

Παράδειγμα 1'

Χρήση των συναρτήσεων append(), capacity(), length() σε ένα αντικείμενο της κλάσης StringBuffer.

```
public class b16
{
    public static void main (String [] args)
    {
        StringBuffer sbf = new StringBuffer (40);
        System.out.println ("sbf = " + sbf);
        System.out.println ("sbf.length () = " + sbf.length ());
    }
}
```

```

System.out.println ("sbf.capacity () = " + sbf.capacity ());

sbf.append ("Athens is ");
System.out.println ("-----");
System.out.println ("sbf = " + sbf);
System.out.println ("sbf.lenght () = " + sbf.length ());
System.out.println ("sbf.capacity () = " + sbf.capacity ());

sbf.append ("the capital of Greece ");
System.out.println ("-----");
System.out.println ("sbf = " + sbf);
System.out.println ("sbf.lenght () = " + sbf.length ());
System.out.println ("sbf.capacity () = " + sbf.capacity ());
}
}

```

Παράδειγμα 2'

Μετατροπή αντικειμένου StringBuffer σε αλφαριθμητικό.

```

public class b18
{
    public static void main (String [] args)
    {
        StringBuffer sbf = new StringBuffer ("my name is george");
        System.out.println ("sbf = " + sbf);
        String str = sbf.toString ();
        System.out.println ("str = " + sbf);
    }
}

```

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1'

Έχουμε το αλφαριθμητικό str1 = "Athens is my favorite city". Ποιο τμήμα του επιστρέφεται με την κλήση της μεθόδου str1.substring(3,6); Εάν δώσουμε str1.substring(4,12), πόσο είναι το μήκος του τμήματος που επιστρέφεται;

Άσκηση 2'

Ποια η κυριότερη διαφορά μεταξύ της κλάσης String και StringBuffer; Με ποιες μεθόδους προσθέτουμε χαρακτήρες στο ήδη

υπάρχον αντικείμενο, βλέπουμε τη συνολική χωρητικότητα και τυπώνουμε το χρησιμοποιούμενο μήκος του αντικειμένου;

Άσκηση 3'

Να γίνει πρόγραμμα, όπου για ένα αντικείμενο της κλάσης StringBuffer, να αλλάξετε κάποιους χαρακτήρες του.

Άσκηση 4'

Να γίνει πρόγραμμα, όπου για ένα αντικείμενο της κλάσης StringBuffer που περιέχει ένα αλφαριθμητικό, να αντιστρέψετε τους χαρακτήρες του.

ΕΝΟΤΗΤΑ 9

Προσδοκώμενα αποτελέσματα

Μετά τη υλοποίηση των δραστηριοτήτων αυτής της ενότητας θα πρέπει να είστε σε θέση :

- 1) Τι είναι η κλάση και τι τα αντικείμενα.
- 2) Ποιος ο σκοπός ύπαρξης των μεταβλητών και των μεθόδων στο ορισμό της κλάσης.
- 3) Τι είναι οι μεταβλητές κλάσης και τι μεταβλητές στιγμιότυπου.
- 4) Τι είναι οι μέθοδοι κλάσης και τι οι μέθοδοι στιγμιότυπου.
- 5) Ποιος ο σκοπός ύπαρξης της μεταβλητής this.
- 6) Τι είναι οι κατασκευαστές, πως ορίζονται, και πως χρησιμοποιούνται για κατασκευάσουμε ένα αντικείμενο.
- 7) Πώς θα χρησιμοποιηθεί η κλάση που κατασκευάσαμε.
- 8) Γιατί υπάρχουν πολλοί κατασκευαστές.
- 9) Τι είναι τα πακέτα, πως χρησιμοποιούμε μία μέθοδο ή μία κλάση ενός πακέτου και ποια τα τυπικά πακέτα της Java.
- 10) Πως μπορούμε να δημιουργήσουμε δικά μας πακέτα, ώστε να βάλουμε μέσα στα πακέτα τις δικές μας κλάσεις.
- 11) Τι είναι η ενθυλάκωση.
- 12) Τι σημαίνει αν οι μεταβλητές και οι μέθοδοι μιας κλάσης δηλωθούν public, private, protected.
- 13) Τι είναι ο πολυμορφισμός και τι κληρονομικότητα.

Ασκήσεις με λυμένα παραδείγματα για εμπέδωση της ύλης

Άσκηση 1΄

- Δημιουργία κλάσης και κατασκευαστή

// κλάσεις και κατασκευαστές

```
class circle
{
    static double pi = 3.14;
    static int count = 0;
    double radius;
    double x;
    double y;
```

```
// δημιουργία του κατασκευαστή.  
  
circle (double r1, double x1, double y1)  
{  
    radius = r1;  
    x = x1;  
    y = y1;  
    ++count;  
}  
  
double area ()  
{  
    return pi * radius * radius;  
}  
}
```

Άσκηση 2'

Πώς θα χρησιμοποιήσουμε τη παραπάνω κλάση που δημιουργήσαμε και πως θα δημιουργηθούν τα αντικείμενα;

Αφού μεταγλωττίσουμε τη παραπάνω κλάση δημιουργείται το *circle.class* αρχείο. Στη συνέχεια γράφουμε ένα άλλο πρόγραμμα , αυτό που ακολουθεί, με το οποίο θα δημιουργήσουμε τα αντικείμενα.

```
public class test  
{  
    public static void main (String [] args)  
    {  
        System.out.println ("number of objects = " + circle.count);  
  
        // δημιουργία αντικειμένου  
  
        // circle = troxos;  
        // troxos = new circle (5.0,1.0,1.0);  
        // οι δύο παραπάνω εντολές με τη επόμενη είναι ίδιες.  
        // Δημιουργούν αντικείμενο  
  
        circle troxos = new circle(5.0, 1.0, 1.0);  
        System.out.println ("number of objects = " + troxos.count);  
  
        circle roda = new circle (4.0, 5.0, 5.0);
```

```

System.out.println ("number of objects = " + roda.count);

// εκτύπωση κάθε αντικειμένου

System.out.println ("area of troxos = " + troxos.area ());
System.out.println ("area of roda = " + roda.area ());
}
}

```

Άσκηση 3'

Άσκηση με πολλαπλούς κατασκευαστές. Στο παρακάτω πρόγραμμα θα δημιουργήσουμε δύο διαφορετικά αντικείμενα, ένα αντικείμενο με το ένα κατασκευαστή και το άλλο αντικείμενο με τον άλλο κατασκευαστή. Και οι δύο κατασκευαστές έχουν το ίδιο όνομα με τη κλάση αλλά διαφορετικές παραμέτρους.

```

// κλάσεις και κατασκευαστές.
// πολλαπλοί κατασκευαστές

class circle1
{
    static double pi = 3.14;
    static int count = 0;
    double radius;
    double x;
    double y;

    // δημιουργία του constructor1

    circle1 (double r1, double x1, double y1)
    {
        radius = r1;
        x = x1;
        y = y1;
        ++count;
    }

    // δημιουργία του costructor2

    circle1 (double r2)
    {

```

```
x = 0.0;
y = 0.0;
radius = r2;
++count;
}

double area()
{
    return pi * radius * radius;
}
}
```

Ακολουθεί το πρόγραμμα που **καλεί τη παραπάνω κλάση** και η δημιουργία των αντικειμένων μέσω πολλαπλών κατασκευαστών

```
public class test
{
    public static void main (String [] args)
    {
        System.out.println ("number of objects = " + circle1.count);

        // δημιουργία του αντικειμένου

        // circle = troxos;
        // troxos = new circle (5.0, 1.0, 1.0);
        // οι δύο παραπάνω εντολές και η ακόλουθη είναι ίδιες
        // καλώντας το πρώτο κατασκευαστή.

        circle1 troxos = new circle1 (5.0, 1.0, 1.0);
        System.out.println ("number of objects = " + troxos.count);

        // καλώντας το δεύτερο constructor

        circle1 roda = new circle1 (4.0);
        System.out.println ("number of objects = " + roda.count);

        // εκτύπωση του εμβαδού για κάθε αντικείμενο που δημιουργήθηκε

        System.out.println ("area of troxos = " + troxos.area ());
        System.out.println ("area of roda = " + roda.area ());
    }
}
```

Άσκηση 4'

Έστω οι παρακάτω εντολές :

```
Circle kiklos1;
```

```
Kiklos1 = new Circle (5.0, 4.0, 5.0);
```

Οι παραπάνω δηλώσεις δημιουργούν στη μνήμη ένα αντικείμενο τύπου Circle με τα χαρακτηριστικά που αναφέρονται και ρυθμίζει τη μεταβλητή kiklos1 ώστε να δείχνει σε αυτό. Μπορούμε επίσης να δημιουργήσουμε και μία άλλη μεταβλητή kiklos2 ως εξής :

```
Circle kiklos2 = kiklos1;
```

Το ερώτημα το οποίο τίθεται είναι το εξής: *Δημιουργήσαμε νέο αντικείμενο; Όχι*, δημιουργήθηκε μία μεταβλητή που δείχνει ακριβώς στο ίδιο αντικείμενο. Εάν κάνουμε μία αλλαγή σε ένα αντικείμενο μέσω μιας μεταβλητής τότε οι αλλαγές περνάνε και στο άλλο αντικείμενο . Αυτό φαίνεται στο πρόγραμμα που ακολουθεί :

```

public class test1
{
    public static void main (String [] args)
    {
        circle kyklos1, kyklos2;
        kyklos1 = new circle (5.0, 4.0, 5.0);
        kyklos2 = kyklos1;
        System.out.println ("kyklos 1 start--> radius = " + kyklos1.radius);
        System.out.println ("kyklos 2 start--> radius = " + kyklos2.radius);
        System.out.println ("-----");
        kyklos2 = kyklos1;
        kyklos1.radius = 20.0;
        System.out.println ("kyklos 1 --> radius = " + kyklos1.radius);
        System.out.println ("kyklos 2 --> radius = " + kyklos2.radius);
        System.out.println ("-----");
        kyklos2.radius = 50.0;
        System.out.println ("kyklos 2 --> radius = " + kyklos2.radius);
        System.out.println ("kyklos 1 --> radius = " + kyklos1.radius);
    }
}

```

Άσκηση 5'**Άσκηση δημιουργίας πακέτου**

Να δημιουργηθεί πακέτο history. Στο πακέτο αυτό να προστεθεί η κλάση circle. Στη συνέχεια να δημιουργηθεί ένα πρόγραμμα το οποίο να κάνει χρήση της κλάσης circle και να εισάγει μέσα στο πρόγραμμα το πακέτο history.

Δημιουργούμε τη public κλάση circle.java και στη κλάση αυτή προσθέτουμε στη αρχή τη λέξη **package** μαζί με το όνομα του πακέτου.

```
package history;
public class circle
{

    double pi = 3.14;
    double radius;
    double x;
    double y;

    // δημιουργία του πρώτου κατασκευαστή.

    public circle (double r1, double x1, double y1) // <--- public
    {
        radius = r1;
        x = x1;
        y = y1;
    }

    // δημιουργία του δεύτερου κατασκευαστή.

    public circle (double r2) // <---- public
    {
        x = 0.0;
        y = 0.0;
        radius = r2;
    }

    public double area() // <--- public
    {
        return pi * radius * radius;
    }
}
```

Στη java δημιουργούμε ένα φάκελο πχ. "classes" κάτω από το φάκελο **c:\jdk1.5.0\jre**. Έπειτα μεταγλωττίζουμε το αρχείο circle.java, έτσι ώστε ο φάκελος history να δημιουργηθεί κάτω από **c:\jdk1.5.0\jre** και στο history υπάρχει circle.class.

Command :---→ **javac -d c:\jdk1.5.0\jre\classes circle.java;**

Έπειτα δημιουργούμε το πρόγραμμα test.java, το οποίο καλεί τη κλάση circle, από το πακέτο history και δημιουργεί τα αντικείμενα troxos και roda της circle

Αφού έχουμε τελειώσει τη μεταγλώττιση καλούμε τη κλάση με τη λέξη **import**. Δημιουργούμε μετά το πρόγραμμα το οποίο καλεί τη κλάση από το πακέτο history και δημιουργεί τα αντικείμενα.

```
import history.circle;
public class test
{
    public static void main (String [] args)
    {
        circle troxos = new circle (5.0, 1.0, 1.0);
        circle roda = new circle (8.0);

        // printing the area of each circle

        System.out.println ("area of troxos = " + troxos.area ());
        System.out.println ("area of roda  = " + roda.area ());
    }
}
```

Άσκηση 6'

Παράδειγμα που επεξηγεί τη κληρονομικότητα των κλάσεων.

```
public class point
{
    protected double x;
    protected double y;
```

```
// ο πρώτος κατασκευαστής
public point (double xcor, double ycor)
{
    x = xcor;
    y = ycor;
}
// ο δεύτερος κατασκευαστής

public point ()
{
    x = 0.0;
    y = 0.0;
}

// ο τρίτος κατασκευαστής

public point (point point1)
{
    x = point1.x;
    y = point1.y;
}

// μέθοδος για τη συντεταγμένη x

public double getx()
{
    return x;
}

// μέθοδος για τη συντεταγμένη y

public double gety()
{
    return y;
}

// μέθοδος για τη μετατροπή της συντεταγμένης του point σε string

public String toString ()
{
    return "(" + x + " , " + y + ")";
}
}
```


Μεταγλωττίζουμε τη παραπάνω κλάση.

Στη συνέχεια δημιουργούμε τη επόμενη κλάση *cycle* ως *υποκλάση* της παραπάνω κλάσης *point*. Η κλάση *cycle* καλεί τη *υπερκλάση* *point* για να δημιουργήσει ένα σημείο στο κέντρο του κύκλου και στη συνέχεια σχεδιάζει τη ακτίνα του. Προσοχή θα πρέπει δοθεί στη δήλωση *extends* καθώς και στη χρήση της εντολής *super*.

```
public class cycle extends point
{
    protected double radius;
    static double pi = 3.14;

    // ο πρώτος κατασκευαστής

    public cycle (double x1, double y1, double r1)
    {
        // η εντολή super καλεί τα μέλη της super class.
        // super () --> καλεί το κατασκευαστή της super class
        // super.f() --> καλεί τη συνάρτηση f(), η οποία έχει οριστεί στη super
            class

        super (x1, y1); // καλεί το κατασκευαστή point της super class cycle
        radius = (r1>=0? r1:0.0); // και περνάμε το x1, y1 για να δώσει
            αρχικές στις μεταβλητές x και y

    // δεύτερος κατασκευαστής
    public cycle ()
    {
        radius = 0.0;
    }

    public double area ()
    {
        return pi * radius *radius;
    }

    public String toString ()
    {
        return "x = " + x + " , " + " y = " + y + " , " + " r = " + radius;
    }
}
```

Δημιουργούμε μία ακόμη υποκλάση της cycle :

```
public class colorcycle extends cycle
{
    protected String color;

    public colorcycle (double x2, double y2, double r2, String c)
    {
        super (x2,y2, r2);
        color = c;
    }

    public String getcolor ()
    {
        return color;
    }

    public String toString()
    {
        return "x = " + x + " , " + " y = " + y + " , " + " r = " + radius +
            " , " + " color = " + color;
    }
}
```

Αποθηκεύουμε και μεταγλωττίζουμε το παραπάνω πρόγραμμα.
Για να δοκιμάσουμε όλα τα παραπάνω δημιουργούμε το παρακάτω πρόγραμμα.

```
public class show
{
    public static void main (String [] args)
    {
        point p1;
        cycle c1;
        colorcycle colorc1;

        p1 = new point (50.5, 70.5);
        c1 = new cycle (40.0, 50.0, 70.5);
        colorc1 = new colorcycle (30.0, 40.0, 20.0,"red");

        // συντεταγμένες του point σε string
        System.out.println ("the point: " + p1.toString ());
    }
}
```

```
// συντεταγμένες του circle c1 και της ακτίνας χωρίς toString ()
System.out.println ("the cycle = " + c1.toString ());

// εμβαδόν της cycle με τη χρήση της area()
System.out.println ("the cycle area = " + c1.area ());

// χωρίς toString() – τα στοιχεία της colorc1
System.out.println ("the colorcycle = " + colorc1.toString ());

// εμβαδόν της colorc1 – χρήση της area() από super class
System.out.println ("the colorcycle.area = "+ colorc1.area ());

// Το χρώμα της cycle με τη χρήση της getcolor
System.out.println ("the colorcycle color = " + colorc1.getcolor ());
}
}
```

Άσκηση 7'

Να γίνει μία κλάση που να ορίζει μία μέθοδο Make. Η μέθοδος να δέχεται δύο ακεραίους, ένα κάτω όριο και ένα άνω όριο, και να δημιουργεί ένα πίνακα που περιέχει όλους τους ακεραίους ανάμεσα στα δύο αυτά όρια.

```
public class Range {
    int [] make(int lower, int upper) {
        int arr [] = new int[ (upper - lower) + 1 ];

        for (int i = 0; i < arr.length; i++) {
            arr [i] = lower++;
        }
        return arr;
    }

    public static void main (String [] arguments) {
        int theArray [];
        Range theRange = new Range ();

        theArray = theRange.make (1, 10);
        System.out.print ("The array: [ ");
        for (int i = 0; i < theArray.length; i++) {
```

```

        System.out.print (theArray[i] + " ");
    }
    System.out.println("");
}
}
}

```

Άσκηση 8'

Παράδειγμα που δείχνει πώς περνάμε παραμέτρους σε μεθόδους.

```

public class PassByReference {
    int onetoZero (int arg []) {
        int count = 0;

        for (int i = 0; i < arg.length; i++) {
            if (arg[i] == 1) {
                count++;
                arg[i] = 0;
            }
        }
        return count;
    }

    public static void main (String[] arguments) {
        int arr [] = { 1, 3, 4, 5, 1, 1, 7 };
        PassByReference test = new PassByReference ();
        int numOnes;

        System.out.print ("Values of the array: [ ");
        for (int i = 0; i < arr.length; i++) {
            System.out.print (arr[i] + " ");
        }
        System.out.println ("]");

        numOnes = test.onetoZero (arr);
        System.out.println ("Number of Ones = " + numOnes);
        System.out.print ("New values of the array: [ ");
        for (int i = 0; i < arr.length; i++) {
            System.out.print (arr[i] + " ");
        }
        System.out.println ("]");
    }
}
}

```

Άσκηση 9'

Το παρακάτω παράδειγμα χρησιμοποιεί **υπερφόρτωση constructors** για να κατασκευάσει τετράγωνα με συντεταγμένες.

```
import java.awt.Point;

class MyRect2 {
    int x1 = 0;
    int y1 = 0;
    int x2 = 0;
    int y2 = 0;

    MyRect2 (int x1, int y1, int x2, int y2) {
        this.x1 = x1;
        this.y1 = y1;
        this.x2 = x2;
        this.y2 = y2;
    }

    MyRect2 (Point topLeft, Point bottomRight) {
        x1 = topLeft.x;
        y1 = topLeft.y;
        x2 = bottomRight.x;
        y2 = bottomRight.y;
    }

    MyRect2 (Point topLeft, int w, int h) {
        x1 = topLeft.x;
        y1 = topLeft.y;
        x2 = (x1 + w);
        y2 = (y1 + h);
    }

    void printRect () {
        System.out.print ("MyRect: <" + x1 + ", " + y1);
        System.out.println (" " + x2 + ", " + y2 + ">");
    }

    public static void main (String [] arguments) {
        MyRect2 rect;

        System.out.println ("Calling MyRect2 with coordinates 25,25
        50,50:");
    }
}
```

```
rect = new MyRect2(25, 25, 50,50);
rect.printRect ();
System.out.println ("***");

System.out.println ("Calling MyRect2 with points (10,10),
(20,20):");
rect= new MyRect2 (new Point(10,10), new Point(20,20));
rect.printRect ();
System.out.println ("***");

System.out.print ("Calling MyRect2 with 1 point (10,10)");
System.out.println (" width (50) and height (50):");
rect = new MyRect2 (new Point(10,10), 50, 50);
rect.printRect ();
System.out.println ("***");

}
}
```

ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ

Άσκηση 1'

Συμπληρώστε τα κενά στις επόμενες προτάσεις :

- A) Ένα αντικείμενο μιαςκλάσης μπορεί να χρησιμοποιηθεί ως αντικείμενο της αντίστοιχης Κλάσης
B) αν η κλάση A κληρονομεί από τη κλάση B τότε η κλάση A λέγεται και η κλάση B λέγεται

Άσκηση 2'

Τι είναι κληρονομικότητα και ποια τα πλεονεκτήματά της; Μπορεί μία κλάση να διαθέτει δύο μεθόδους με το ίδιο όνομα οι οποίες να επιστρέφουν διαφορετικούς τύπους δεδομένων;

Άσκηση 3'

Τι είναι κλάση στη Java . Ποια είναι τα βασικά μέρη από τα οποία αποτελείται μία άδεια κλάση; Πως μπορούμε με τη βοήθεια των κλάσεων να οργανώσουμε καλύτερα τα προγράμματα μας ;

Άσκηση 4'

Ποια είναι τα πλεονεκτήματα του αντικειμενοστραφούς προγραμματισμού; Με ποιόν τρόπο δημιουργούμε ένα αντικείμενο μιας κλάσης; Πως δημιουργούμε μια υποκλάση;

Άσκηση 5'

Οι κατασκευαστές μιας κλάσης ονομάζονται ..
Α) main β) extends γ) constructor δ) object.

Άσκηση 6'

Ο τύπος δεδομένων που επιστρέφει ο κατασκευαστής μιας κλάσης είναι :
Α) τύπου float β) ίδιος με τον τύπο της κλάσης γ) τύπου int δ) οι κατασκευαστές δεν επιστρέφουν κανένα τύπο.

Δημιουργική Άσκηση

Οι μαθητές, αφού χωριστούν σε ομάδες τριών ατόμων, να υλοποιήσουν την παρακάτω άσκηση που αφορά κλάσεις και κληρονομικότητες.

Δημιουργήστε μία κλάση με το όνομα Clock με τα ακόλουθα χαρακτηριστικά :

Ωρα → Hour (ακέραιος από 1...12)
Λεπτά → Min (ακέραιος από 0...59)
Δευτ/τα → Sec (ακέραιος από 0...59)
Είναι Π.Μ. → isAM (Boolean)

Η κλάση Clock υποστηρίζει τις παρακάτω μεθόδους :

- getHours()
- getMinutes()
- getSeconds()
- getIsAM()

Η κλάση θα έχει τους εξής κατασκευαστές :

- Έναν χωρίς καθόλου παραμέτρους, ο οποίος να θέτει το ρολόι στα μεσάνυχτα.
- Έναν με παραμέτρους την ώρα και την ένδειξη ή όχι “π.μ.”.
- Έναν με όλες τις παραμέτρους.
- Η κλάση Clock να υποστηρίζει επίσης μία μέθοδο setTime(), η οποία να θέτει όλα τα χαρακτηριστικά σε προκαθορισμένες τιμές.



ΥΠΟΥΡΓΕΙΟ ΕΘΝΙΚΗΣ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ ΕΠΕΑΕΚ

ΕΥΡΩΠΑΪΚΗ ΕΝΩΣΗ
ΣΥΓΧΡΗΜΑΤΟΔΟΤΗΣΗ
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ



Η ΠΑΙΔΕΙΑ ΣΤΗΝ ΚΟΡΥΦΗ
Επιχειρησιακό Πρόγραμμα
Εκπαίδευσης και Αρχικής
Επαγγελματικής Κατάρτισης