

## **Κεφάλαιο II:**

### **Εργαστηριακές ασκήσεις που αφορούν βασικές έννοιες, τύπους δεδομένων και συμβολοσειρές στη γλώσσα προγραμματισμού Java.**

Στο παρόν κεφάλαιο παρουσιάζονται εργαστηριακές ασκήσεις οι οποίες αφορούν βασικές έννοιες, τύπους δεδομένων και συμβολοσειρές στη γλώσσα προγραμματισμού Java. Παρουσιάζονται ασκήσεις οι οποίες αναφέρονται σε:

- **Μεταβλητές**
- **Τελεστές**
- **Τύπους δεδομένων**
- **Συμβολοσειρές**
- **Στις μεθόδους `println()` και `print()`**
- **Στις μαθηματικές μεθόδους**
- **Απλά υπολογιστικά προβλήματα φυσικής**
- **Εισαγωγή δεδομένων σε προγράμματα**

### i) Οι βασικοί τύποι δεδομένων στην Java.

Οι βασικοί τύποι δεδομένων στη Java είναι οκτώ και εικονίζονται στον επόμενο πίνακα

Τύπος	Μέγεθος σε bits	Τιμές
byte	8	-128 έως 127
short	16	-32768 έως 32767
int	32	-2146473648 έως 2147483647
long	64	-9223372036854775808 έως 9223372036854775807
float	32	$\pm 1.40129846432481707 \times 10^{-45}$ έως $\pm 3.40282346638528860 \times 10^{38}$
double	64	$\pm 4.94065645841246544 \times 10^{-324}$ έως $\pm 1.79769313486231570 \times 10^{308}$
char	16	Όλοι οι Unicode χαρακτήρες
boolean		false ή true

### ii) Οι βασικοί τελεστές στη Java.

Στην Java ορίζονται οι παρακάτω **αριθμητικοί τελεστές** οι οποίοι μας επιτρέπουν να κάνουμε πράξεις:

Τελεστής	Περιγραφή
+	Πρόσθεση
-	Αφαίρεση
*	Πολλαπλασιασμός
/	Διαίρεση
%	Υπόλοιπο

Εκτός από τους αριθμητικούς τελεστές συναντάμε και τους ακόλουθους βασικούς τελεστές:

- Οι **συσχετιστικοί τελεστές** είναι οι ακόλουθοι:

Τελεστής	Περιγραφή
>	Μεγαλύτερο
>=	Μεγαλύτερο ή ίσο
<	Μικρότερο
<=	Μικρότερο ή ίσο

- Οι τελεστές ισότητας είναι οι ακόλουθοι:

Τελεστής	Περιγραφή
==	Ίσο με
!=	Άνισο με

- Οι λογικοί τελεστές είναι οι ακόλουθοι:

Τελεστής	Περιγραφή
&&	Λογικός τελεστής AND
	Λογικός τελεστής OR
!	Λογικός τελεστής NEGATION

Τους συσχετιστικούς τελεστές, τους τελεστές ισότητας και τους λογικούς τελεστές τους συναντάμε κυρίως στις εντολές **if**, **for**, **while**, **do**. Οι παραπάνω τελεστές χρησιμοποιούνται για συγκρίσεις μεταξύ αριθμών, μεταβλητών και παραστάσεων. Εάν η σύγκριση είναι **αληθής** τότε το αποτέλεσμα είναι **true** διαφορετικά εάν είναι **ψευδής** τότε το αποτέλεσμα είναι **false**.

- Ο τελεστής αύξησης και ο τελεστής μείωσης είναι οι ακόλουθοι:

Τελεστής	Περιγραφή
++	Τελεστής αύξησης κατά 1
--	Τελεστής μείωσης κατά 1

Οι τελεστές ++ και – χρησιμοποιούνται όταν θέλουμε να προσθέσουμε ή να αφαιρέσουμε το 1 από μία μεταβλητή. Έτσι το ++a; ισοδυναμεί με το a=a+1; ενώ το –a; ισοδυναμεί στο a=a-1; Εδώ πρέπει να σημειώσουμε πως οι τελεστές ++ και – μπορούν να χρησιμοποιηθούν είτε ως **προθεματικοί** τελεστές (δηλ. πριν την μεταβλητή, όπως ++a ή –a) είτε ως **μεταθεματικοί** (δηλ. μετά την μεταβλητή, όπως a++ ή a–). Και στις δύο περιπτώσεις η τιμή του a αυξάνει ή ελαττώνεται κατά 1. Για παράδειγμα στην παράσταση ++a η τιμή του a αυξάνει πριν χρησιμοποιηθεί η τιμή της, ενώ στην παράσταση a++ η τιμή του a αυξάνει αφού χρησιμοποιηθεί η τιμή της. Έτσι έστω ότι το a ισούται με 5 τότε η παράσταση b = a++; δίνει στο b την τιμή 5 ενώ η παράσταση b=++a; την τιμή 6. Το a και στις δύο περιπτώσεις γίνεται 6.

- Ο τελεστής αντιστοίχισης είναι ο:

Τελεστής	Περιγραφή
=	Τελεστής αντιστοίχισης

- οι τελεστές αντικατάστασης είναι οι ακόλουθοι

Τελεστής	Περιγραφή
+=	Τελεστής πρόσθεσης και αντιστοίχισης
-=	Τελεστής αφαίρεσης και αντιστοίχισης
*=	Τελεστής πολ/μου και αντιστοίχισης
/=	Τελεστής διαίρεσης και αντιστοίχισης
%=	Τελεστής υπολοίπου και αντιστοίχισης

Ο τελεστής = αντιστοιχεί την δεξιά τιμή μιας παράστασης στην αριστερή. Έτσι η παράσταση  
a=b;

αντιστοιχεί την τιμή του b στο a. Τα παρακάτω παραδείγματα δίνουν την έννοια των υπολοίπων από τους παραπάνω τελεστές:

Το a += b; ισοδυναμεί με το a = a+b;  
 Το a -= b; ισοδυναμεί με το a = a-b;  
 Το a \*= b; ισοδυναμεί με το a = a\*b;  
 Το a /= b; ισοδυναμεί με το a = a/b;  
 Το a %= b; ισοδυναμεί με το a = a %b;

- Οι τελεστές πράξεων με bits είναι οι ακόλουθοι :

Τελεστής	Περιγραφή
&	AND για bit
	OR για bit
^	XOR για bit
~	NOT για bit
>>	Ολίσθηση αριστερά
<<	Ολίσθηση δεξιά

Οι παραπάνω τελεστές αφορούν πράξεις σε επίπεδο bits και για να τους καταλάβουμε πρέπει να διαχειριζόμαστε τους διάφορους αριθμούς σε δυαδική μορφή. Οι τελεστές &, |, ^ και ~ αντιστοιχούν στις απλές πράξεις της **άλγεβρας Boole**. Οι τελεστές >> και << προκαλούν ολίσθηση στα δεξιά και στα αριστερά αντίστοιχα. Έτσι για παράδειγμα εάν η μεταβλητή a είναι ο δυαδικός αριθμός 01101000 τότε η παράσταση

b = a >> 2;

δίνει στη μεταβλητή b την τιμή 00011010.

### iii) Η κλάση Math στη Java.

Η κλάση Math περιέχει μεθόδους με τις οποίες μπορούμε να κάνουμε βασικές πράξεις με εκθετικά, λογαριθμούς, τετραγωνικές ρίζες και τριγωνομετρικές συναρτήσεις. Παραδείγματα:

double c=Math.pow(a,b)      αντιστοιχεί σε  $c=a^b$   
double b=Math.cos(a)      αντιστοιχεί σε  $b=\cos(a)$   
double b=Math.abs(a)      αντιστοιχεί σε  $b=|a|$

Στη συνέχεια παραθέτουμε τα περιεχόμενα της ιστοσελίδας <http://java.sun.com/j2se/1.4.2/docs/api/java/lang/Math.html> που αφορά την κλάση Math. Οι φοιτητές μπορούν να ανατρέξουν σε αυτή για περισσότερες πληροφορίες.

Field Summary	
static double	<a href="#"><u>E</u></a> The double value that is closer than any other to $e$ , the base of the natural logarithms.
static double	<a href="#"><u>PI</u></a> The double value that is closer than any other to $pi$ , the ratio of the circumference of a circle to its diameter.

Method Summary	
static double	<a href="#"><u>abs</u></a> (double a) Returns the absolute value of a double value.
static float	<a href="#"><u>abs</u></a> (float a) Returns the absolute value of a float value.
static int	<a href="#"><u>abs</u></a> (int a) Returns the absolute value of an int value.
static long	<a href="#"><u>abs</u></a> (long a) Returns the absolute value of a long value.
static double	<a href="#"><u>acos</u></a> (double a) Returns the arc cosine of an angle, in the range of 0.0 through $pi$ .
static double	<a href="#"><u>asin</u></a> (double a) Returns the arc sine of an angle, in the range of $-pi/2$ through $pi/2$ .
static double	<a href="#"><u>atan</u></a> (double a) Returns the arc tangent of an angle, in the range of $-pi/2$ through $pi/2$ .
static double	<a href="#"><u>atan2</u></a> (double y, double x) Converts rectangular coordinates (x, y) to polar (r, theta).

static double	<a href="#"><u>ceil</u></a> (double a) Returns the smallest (closest to negative infinity) double value that is not less than the argument and is equal to a mathematical integer.
static double	<a href="#"><u>cos</u></a> (double a) Returns the trigonometric cosine of an angle.
static double	<a href="#"><u>exp</u></a> (double a) Returns Euler's number $e$ raised to the power of a double value.
static double	<a href="#"><u>floor</u></a> (double a) Returns the largest (closest to positive infinity) double value that is not greater than the argument and is equal to a mathematical integer.
static double	<a href="#"><u>IEEERemainder</u></a> (double f1, double f2) Computes the remainder operation on two arguments as prescribed by the IEEE 754 standard.
static double	<a href="#"><u>log</u></a> (double a) Returns the natural logarithm (base $e$ ) of a double value.
static double	<a href="#"><u>max</u></a> (double a, double b) Returns the greater of two double values.
static float	<a href="#"><u>max</u></a> (float a, float b) Returns the greater of two float values.
static int	<a href="#"><u>max</u></a> (int a, int b) Returns the greater of two int values.
static long	<a href="#"><u>max</u></a> (long a, long b) Returns the greater of two long values.
static double	<a href="#"><u>min</u></a> (double a, double b) Returns the smaller of two double values.
static float	<a href="#"><u>min</u></a> (float a, float b) Returns the smaller of two float values.
static int	<a href="#"><u>min</u></a> (int a, int b) Returns the smaller of two int values.
static long	<a href="#"><u>min</u></a> (long a, long b) Returns the smaller of two long values.
static double	<a href="#"><u>pow</u></a> (double a, double b) Returns of value of the first argument raised to the power of the second argument.
static double	<a href="#"><u>random</u></a> () Returns a double value with a positive sign, greater than or equal to 0.0 and less than 1.0.
static double	<a href="#"><u>rint</u></a> (double a) Returns the double value that is closest in value to the argument and is equal to a mathematical integer.
static long	<a href="#"><u>round</u></a> (double a)

	Returns the closest <code>long</code> to the argument.
<code>static int</code>	<a href="#"><u>round</u></a> (float a) Returns the closest <code>int</code> to the argument.
<code>static double</code>	<a href="#"><u>sin</u></a> (double a) Returns the trigonometric sine of an angle.
<code>static double</code>	<a href="#"><u>sqrt</u></a> (double a) Returns the correctly rounded positive square root of a <code>double</code> value.
<code>static double</code>	<a href="#"><u>tan</u></a> (double a) Returns the trigonometric tangent of an angle.
<code>static double</code>	<a href="#"><u>toDegrees</u></a> (double angrad) Converts an angle measured in radians to an approximately equivalent angle measured in degrees.
<code>static double</code>	<a href="#"><u>toRadians</u></a> (double angdeg) Converts an angle measured in degrees to an approximately equivalent angle measured in radians.

## 2.1 Λυμένες Ασκήσεις.

- 2.1.1 Γράψτε ένα πρόγραμμα το οποίο να εμφανίζει τις παρακάτω γραμμές  
*The Java programming language was developed by  
James Gosling at Sun Microsystems in 1991.  
When the World Wide Web appeared on the Internet in 1993,  
the language was enhanced to facilitate programming on the web.*

χρησιμοποιώντας την μέθοδο `println()` και ένα δεύτερο χρησιμοποιώντας την μέθοδο `print()`.

Μια πιθανή λύση του προβλήματος χρησιμοποιώντας την μέθοδο `println()` είναι η ακόλουθη:

```
class Erg211a
{
    public static void main(String[] arguments)
    {
        System.out.println("The Java programming language was developed by");
        System.out.println("James Gosling at Sun Microsystems in 1991.");
        System.out.println("When the World Wide Web appeared on the Internet in 1993,");
        System.out.println("the language was enhanced to facilitate programming on the web.");
    }
}
--:-- Erg211a.java (Java Abbrev)--L10--A11-----
```

Η εκτέλεση και τα αποτελέσματα του παραπάνω προγράμματος εικονίζονται στο παρακάτω σχήμα:

```
[student1@pc244 kef2]$
[student1@pc244 kef2]$
[student1@pc244 kef2]$ javac Erg211a.java
[student1@pc244 kef2]$ java Erg211a
The Java programming language was developed by
James Gosling at Sun Microsystems in 1991.
When the World Wide Web appeared on the Internet in 1993,
the language was enhanced to facilitate programming on the web.
[student1@pc244 kef2]$ □
```

Μια πιθανή λύση του προβλήματος χρησιμοποιώντας την μέθοδο `print()` είναι η ακόλουθη:

```
class Erg211b
{
    public static void main(String[] arguments)
    {
        System.out.print("The Java programming language was developed by\n");
        System.out.print("James Gosling at Sun Microsystems in 1991.\n");
        System.out.print("When the World Wide Web appeared on the Internet in 1993,\n");
        System.out.print("the language was enhanced to facilitate programming on the web.\n");
    }
}
--:-- Erg211b.java (Java Abbrev)--L10--A11-----
```



Η εκτέλεση και τα αποτελέσματα του παραπάνω προγράμματος εικονίζονται στο παρακάτω σχήμα:

```
[student1@pc244 kef2]$  
[student1@pc244 kef2]$  
[student1@pc244 kef2]$ javac Erg211b.java  
[student1@pc244 kef2]$ java Erg211b  
The Java programming language was developed by  
James Gosling at Sun Microsystems in 1991.  
When the World Wide Web appeared on the Internet in 1993,  
the language was enhanced to facilitate programming on the web.  
[student1@pc244 kef2]$ □
```

Όπως βλέπουμε οι εκτελέσεις των δύο προγραμμάτων δίνουν το ίδιο αποτέλεσμα. Η διαφορά μεταξύ των μεθόδων `println()` και `print()` είναι ο χαρακτήρας νέας γραμμής `\n`.

**2.1.2** Γράψτε ένα πρόγραμμα στο οποίο να δηλώσετε 8 μεταβλητές σύμφωνα με τους οκτώ βασικούς τύπους δεδομένων που συναντάμε στην Java. Δώστε τιμές σε αυτές τις μεταβλητές και στη συνέχεια τυπώστε τις.

Μια πιθανή λύση του προβλήματος είναι η ακόλουθη:

```
class PrintType  
{  
    public static void main(String[] arguments)  
    {  
        byte i=127;  
        short j=32767;  
        int k=2147483647;  
        long l=9223372036854775807L; // To 'L' mpainei gia na dilosei ton long  
  
        float x=3.14159265F;          // To 'F' mpainei gia na dilosei ton float  
        double y=3.141592653589793238;  
  
        char c='A';  
        boolean b=false;  
  
        System.out.println("i = " + i);  
        System.out.println("j = " + j);  
        System.out.println("k = " + k);  
        System.out.println("l = " + l);  
        System.out.println("x = " + x);  
        System.out.println("y = " + y);  
        System.out.println("c = " + c);  
        System.out.println("b = " + b);  
    }  
}
```

-- PrintType.java (Java Abbrev)--L25--All-----

Η εκτέλεση και τα αποτελέσματα του παραπάνω προγράμματος εικονίζονται στο παρακάτω σχήμα:

```

[student1@pc244 kef2]$ javac PrintType.java
[student1@pc244 kef2]$ java PrintType
i = 127
j = 32767
k = 2147483647
l = 9223372036854775807
x = 3,1415927
y = 3,141592653589793
c = A
b = false
[student1@pc244 kef2]$ █

```

Προσοχή όταν χρησιμοποιούμε αριθμούς τύπου long και float πρέπει να τερματίζονται κατά την αρχικοποίηση από το γράμμα ‘L’ και ‘F’ αντίστοιχα.

**2.1.3** Γράψτε ένα πρόγραμμα το οποίο να υπολογίζει και να εκτυπώνει το συνημίτονο, το ημίτονο και την εφαπτομένη των  $60^\circ$ .

Μια πιθανή λύση είναι η ακόλουθη:

```

class Erg213
{
    public static void main(String[] arguments)
    {
        double theta=60.;           // H theta se moires
        theta=theta*Math.PI/180.;   // H theta se rads
        System.out.println("To cos ton 60 moiron einai : " + Math.cos(theta));
        System.out.println("To sin ton 60 moiron einai : " + Math.sin(theta));
        System.out.println("H tan ton 60 moiron einai : " + Math.tan(theta));
    }
}

```

--- Erg213.java (Java Abbrev)--L11--Top-----

Η εκτέλεση και τα αποτελέσματα του παραπάνω προγράμματος εικονίζονται στο παρακάτω σχήμα:

```

[student1@pc244 kef2]$ javac Erg213.java
[student1@pc244 kef2]$ java Erg213
To cos ton 60 moiron einai : 0,5000000000000001
To sin ton 60 moiron einai : 0,8660254037844386
H tan ton 60 moiron einai : 1,7320508075688767
[student1@pc244 kef2]$ █

```

Το όρισμα στις τριγωνομετρικές μεθόδους πρέπει να εκφράζεται σε rads.

**2.1.4** Ένα σώμα αφήνεται να πέσει ελεύθερα από ηρεμία. Συντάξτε ένα πρόγραμμα το οποίο να υπολογίζει την απόσταση που διανύει το σώμα και την ταχύτητά του μετά από 1 sec και μετά από 5 sec. (Δίνονται  $s=1/2*(g*t^2)$ ,  $u=g*t$ ,  $g=9.81 \text{ m/sec}^2$ )

Μια πιθανή λύση είναι η ακόλουθη:

```
class FreeFall
{
    public static void main(String[] arguments)
    {
        double g=9.81;

        double xronos=1.;
        double apostasi=0.5*g*Math.pow(xronos,2);
        double taxitita=g*xronos;
        System.out.println("Gia xrono= " + xronos + " sec, apostasi = " + apostasi + " m");
        System.out.println("Gia xrono= " + xronos + " sec, taxitita = " + taxitita + " m/sec");

        xronos=5.;
        apostasi=0.5*g*Math.pow(xronos,2);
        taxitita=g*xronos;
        System.out.println("Gia xrono= " + xronos + " sec, apostasi = " + apostasi + " m");
        System.out.println("Gia xrono= " + xronos + " sec, taxitita = " + taxitita + " m/sec");
    }
}
FreeFall.java (Java Abbrev)--L20--A11-----
```

Η εκτέλεση του παραπάνω προγράμματος παράγει το ακόλουθο αποτέλεσμα:

```
[student1@pc244 kef2]$
[student1@pc244 kef2]$
[student1@pc244 kef2]$ javac FreeFall.java
[student1@pc244 kef2]$ java FreeFall
Gia xrono= 1.0 sec, apostasi = 4.905 m
Gia xrono= 1.0 sec, taxitita = 9.81 m/sec
Gia xrono= 5.0 sec, apostasi = 122.625 m
Gia xrono= 5.0 sec, taxitita = 49.050000000000004 m/sec
[student1@pc244 kef2]$
```

**2.1.5** Γράψτε ένα πρόγραμμα στο οποίο να ορίσετε μία συμβολοσειρά και να αποθηκεύσετε σε αυτή την φράση *University of Ioannina*. Τυπώστε την συμβολοσειρά και τον αριθμό των χαρακτήρων που περιέχει. Κατασκευάστε μια νέα συμβολοσειρά που να περιέχει την προηγούμενη και επιπλέον την φράση *Physics Department*. Τυπώστε την νέα συμβολοσειρά και τον αριθμό των χαρακτήρων που περιέχει. Δημιουργήστε από τις παραπάνω συμβολοσειρές δύο νέες οι οποίες να περιέχουν τις ίδιες φράσεις στα κεφαλαία. Τυπώστε τις.

Μια πιθανή λύση είναι η ακόλουθη:

```
class Uoi
{
    public static void main(String[] arguments)
    {
        String uoi="University of Ioannina";
        System.out.println(uoi);
        System.out.println("Arithmos xaraktiron uoi : " + uoi.length());

        String uoipd=uai + " Physics Department"; // epikollisi simvolosiron
        System.out.println(uoipd);
        System.out.println("Arithmos xaraktiron uoipd: " + uoipd.length());

        String UOI=uai.toUpperCase(); // metatropi se kefalaia
        String UOIPD=uoipd.toUpperCase(); // metatropi se kefalaia
        System.out.println(UOI);
        System.out.println(UOIPD);
    }
}

--:-- Uoi.java (Java Abbrev)--L19--All-----
```

Η εκτέλεση του παραπάνω προγράμματος παράγει το ακόλουθο αποτέλεσμα:

```
[student1@pc244 kef2]$ javac Uoi.java
[student1@pc244 kef2]$ java Uoi
University of Ioannina
Arithmos xaraktiron uoi : 22
University of Ioannina Physics Department
Arithmos xaraktiron uoipd: 41
UNIVERSITY OF IOANNINA
UNIVERSITY OF IOANNINA PHYSICS DEPARTMENT
[student1@pc244 kef2]$ □
```

Στο παραπάνω πρόγραμμα χρησιμοποιήσαμε την μέθοδο **toUpperCase()** για την μετατροπή από πεζά σε κεφαλαία. Ομοίως υπάρχει και η μέθοδος **toLowerCase()** που κάνει το αντίστροφο. Περισσότερες πληροφορίες για την κλάση `String` θα βρείτε στην ιστοσελίδα: <http://java.sun.com/j2se/1.4.2/docs/api/java/lang/String.html>

**2.1.6** Γράψτε ένα πρόγραμμα στο οποίο να εισάγετε από την γραμμή εντολών ένα string έναν int και έναν double. Στη συνέχεια να τα τυπώσετε.

Μια πιθανή λύση είναι η ακόλουθη:

```
class InLine
{
    public static void main(String[] arguments)
    {
        String mystring=arguments[0];
        int myint=Integer.parseInt(arguments[1]);
        double mydouble=Double.parseDouble(arguments[2]);

        System.out.println("Mystring = " + mystring);
        System.out.println("Myint    = " + myint);
        System.out.println("Mydouble = " + mydouble);
    }
}
```

-- InLine.java (Java Abbrev)--L14--All-----

Η εκτέλεση του παραπάνω προγράμματος παράγει το ακόλουθο αποτέλεσμα όταν στη γραμμή εντολών δακτυλογραφηθούν η συμβολοσειρά **Hello** ο ακέραιος **123** και ο αριθμός κινητής υποδιαστολής διπλής ακρίβειας **3.14159**:

```
[student1@pc244 kef2]$ javac InLine.java
[student1@pc244 kef2]$ java InLine Hello 123 3.14159
Mystring = Hello
Myint    = 123
Mydouble = 3.14159
[student1@pc244 kef2]$ []
```

Παρατηρούμε πως αυτός είναι ένας απλός τρόπος που μπορούμε να χρησιμοποιήσουμε για να εισάγουμε δεδομένα στο πρόγραμμά μας. Εκμεταλλευόμαστε τον πίνακα συμβολοσειράς `String[] arguments` ο οποίος αποτελεί το όρισμα της μεθόδου `main()`. Τα `Hello`, `123` και `3.14159` εισάγονται ως συμβολοσειρές στο πρόγραμμα και αποθηκεύονται στα πρώτα στοιχεία του πίνακα `arguments` (δηλαδή το `Hello` στο `arguments[0]`, το `123` στο `arguments[1]` και το `3.14159` στο `arguments[2]`).

Στη συνέχεια για να μετατρέψουμε την συμβολοσειρά `123` σε ακέραιο χρησιμοποιούμε όπως φαίνεται στο παράδειγμα την μέθοδο `parseInt()` της κλάσης `Integer`. Για να μετατρέψουμε την συμβολοσειρά `3.14159` σε αριθμός κινητής υποδιαστολής διπλής ακρίβειας χρησιμοποιούμε όπως φαίνεται στο παράδειγμα την μέθοδο `parseDouble()` της κλάσης `Double`. Περισσότερα για τις κλάσεις `Integer` και `Double` μπορείτε να βρείτε στις ιστοσελίδες:

<http://java.sun.com/j2se/1.4.2/docs/api/java/lang/Integer.html>

<http://java.sun.com/j2se/1.4.2/docs/api/java/lang/Double.html>

2.1.7 Να γράψετε ένα πρόγραμμα το οποίο να υπολογίζει την αριθμητική τιμή της συνάρτησης  $f(x) = \sqrt{\frac{4x^3 - 3x^2 + 2x - 1}{5}}$  για διάφορες τιμές του x. Οι τιμές του x να εισάγονται στο πρόγραμμα από την γραμμή εντολών. Εκτελέστε το πρόγραμμα δίνοντας τέσσερα δικά σας παραδείγματα.

Μια πιθανή λύση είναι η ακόλουθη:

```
// To programma auto dexete enan double apo thn grammi entolon
class Fx
{
    public static void main(String[] arguments)
    {
        double x=Double.parseDouble(arguments[0]);
        double fx=Math.sqrt((4.*Math.pow(x,3)-3.*Math.pow(x,2)+2.*x-1)/5.);
        System.out.println("Gia x = " + x + " f(x) = " + fx);
    }
}

:-- Fx.java (Java Abbrev)--L11--A11-----
```

Η εκτέλεση του παραπάνω προγράμματος με τέσσερα τυχαία παραδείγματα είναι η ακόλουθη:

```
[student1@pc244 kef2]$ javac Fx.java
[student1@pc244 kef2]$ java Fx 2,55
Gia x = 2,55 f(x) = 3,191175332068108
[student1@pc244 kef2]$
[student1@pc244 kef2]$ java Fx 8,94
Gia x = 8,94 f(x) = 22,957252170066
[student1@pc244 kef2]$
[student1@pc244 kef2]$ java Fx 17,9
Gia x = 17,9 f(x) = 66,35499378343727
[student1@pc244 kef2]$
[student1@pc244 kef2]$ java Fx -34,789
Gia x = -34,789 f(x) = NaN
[student1@pc244 kef2]$
```

Παρατηρείστε πως για την τιμή  $x=-34.789$  η τιμή της παράστασης δεν ορίζεται γιατί το υπόριζο είναι αρνητικό. Γι αυτό το λόγο εκτυπώνεται το NaN. Σε τέτοιες περιπτώσεις πρέπει να προστατεύουμε το πρόγραμμα όπως θα δούμε αργότερα με την εντολή if.

**2.1.8** Να γράψετε ένα πρόγραμμα στο οποίο να εισάγετε το όνομά σας την ηλικία σας και το τρέχον έτος και να σας επιστρέφει το έτος γέννησης σας. Η εισαγωγή των δεδομένων να γίνει από το πληκτρολόγιο.

Στα προηγούμενα δύο παραδείγματα μάθαμε πως να εισάγουμε δεδομένα στο πρόγραμμά μας από την γραμμή εντολών. Ένας άλλος τρόπος εισαγωγής δεδομένων στο πρόγραμμα περιγράφεται σε αυτό το παράδειγμα. Το πρόγραμμα κατά την εκτέλεση σταματά και περιμένει την εισαγωγή των δεδομένων. Αυτός είναι πιο σωστός τρόπος αλληλεπίδρασης προγράμματος-χρήστη. Μια πιθανή λύση στο πρόβλημα είναι η ακόλουθη:

```
import java.io.*;

class BirthYear
{
    public static void main(String[] arguments) throws IOException
    {
        InputStreamReader reader = new InputStreamReader(System.in);
        BufferedReader input = new BufferedReader(reader);

        System.out.print("Dose to onoma sou : ");
        String onoma = input.readLine();           // Eisagogi tou onomatos

        System.out.print("Dose tin ilikia sou : ");
        String ilikia = input.readLine();         // Eisagogi tis ilikias
        int xronia = Integer.parseInt(ilikia);    // metaropi ilikias se akeraio

        System.out.print("Dose to trexon etos: ");
        String trexon_etos = input.readLine();    // Eisagogi tou trexontos etous
        int etos = Integer.parseInt(trexon_etos); // metaropi trexontos etous se akeraio

        int etos_genisis = etos - xronia;        // ypologismos etous genisis

        System.out.println("To onoma sou einai : " + onoma);
        System.out.println("Eisai : " + xronia + " eton.");
        System.out.println("Genithikes to : " + etos_genisis);
    }
}

:.* BirthYear.java (Java Abbrev)--L28--A11-----
```

Η εκτέλεση του παραπάνω προγράμματος είναι η ακόλουθη:

```
[student1@pc244 kef2]$ javac BirthYear.java
[student1@pc244 kef2]$ java BirthYear
Dose to onoma sou : Dimitris
Dose tin ilikia sou : 21
Dose to trexon etos: 2003
To onoma sou einai : Dimitris
Eisai : 21 eton.
Genithikes to : 1982
[student1@pc244 kef2]$
```

Το αντικείμενο reader της κλάσης **InputStreamReader** διαβάζει τα δεδομένα από το πληκτρολόγιο. Το αντικείμενο input της κλάσης **BufferedReader** μετατρέπει κάθε byte πληροφορίας από το αντικείμενο reader σε χαρακτήρες. Η μέθοδος **readline()** μεταφέρει τα δεδομένα που δακτυλογραφούνται στις συμβολοσειρές μας (π.χ onoma, ilikia, trexon\_etos) για περαιτέρω επεξεργασία. Τέλος το “**throws IOException**” είναι απαραίτητο για να χρησιμοποιηθεί η μέθοδος **readline()**.

**2.1.9** Γράψτε ένα πρόγραμμα το οποίο να επαληθεύει την ισότητα:

$$\sin(A+B) = \sin A \cos B + \cos A \sin B$$

Να υπολογίσετε ξεχωριστά το κάθε μέλος της ισότητας και να το εκτυπώσετε για τιμές των μεταβλητών A και B που εισάγονται από το πληκτρολόγιο. Δώστε ένα παραδείγματα.

Μια πιθανή λύση είναι η ακόλουθη:

```
import java.io.*;

class SinCos
{
    public static void main(String[] arguments) throws IOException
    {
        InputStreamReader reader = new InputStreamReader(System.in);
        BufferedReader input = new BufferedReader(reader);

        System.out.print("Dose tin gonias A se moires : ");
        String goniasA = input.readLine(); // Eisagogi gonias A
        double A = Double.parseDouble(goniasA); // metaropi gonias A se double
        A *= Math.PI/180.; // metaropi se rads

        System.out.print("Dose tin gonias B se moires : ");
        String goniasB = input.readLine(); // Eisagogi gonias B
        double B = Double.parseDouble(goniasB); // metaropi gonias B se double
        B *= Math.PI/180.; // metaropi se rads

        double left_side = Math.sin(A+B);
        double right_side = Math.sin(A)*Math.cos(B)+Math.cos(A)*Math.sin(B);

        System.out.println("Gia gvnies A=" + goniasA + " kai B=" + goniasB + " exoume :");
        System.out.println("sin(A+B) = " + left_side);
        System.out.println("sin(A)cos(B) + cos(A)sin(B) = " + right_side);
    }
}

-- SinCos.java (Java Abbrev)--L27--All-----
```

Η εκτέλεση του παραπάνω προγράμματος με τρία τυχαία παραδείγματα είναι η ακόλουθη:

```
[student1@pc244 kef2]$ javac SinCos.java
[student1@pc244 kef2]$ java SinCos
Dose tin gonias A se moires : 45
Dose tin gonias B se moires : 60
Gia gvnies A=45 kai B=60 exoume :
sin(A+B) = 0.9659258262890683
sin(A)cos(B) + cos(A)sin(B) = 0.9659258262890684
[student1@pc244 kef2]$
```



## 2.2 Ασκήσεις.

**2.2.1** Ένα σώμα εκτελεί πλάγια βολή. Το σώμα εκτοξεύεται με αρχική ταχύτητα  $v_0=100\text{m/sec}$ . Να γράψετε ένα πρόγραμμα το οποίο να υπολογίζει για πόσο χρόνο το σώμα μένει στον αέρα και ποιο είναι το βεληνεκές του όταν εκτοξεύεται υπό γωνία  $\theta$  ίση με  $30^\circ$ ,  $40^\circ$ ,  $45^\circ$ ,  $50^\circ$  και  $60^\circ$ . Αγνοείστε την τριβή του αέρα. Στα αποτελέσματά σας να αναγράψετε και μονάδες. Δίνονται:  $g=9.81\text{m/sec}^2$ ,  $t=(2 v_0 \sin\theta)/(g)$  και  $R=(v_0 \cos\theta)t$

**2.2.2** Να αναπτύξετε προγράμματα τα οποία να επαληθεύουν κάθε μία από τις ακόλουθες τριγωνομετρικές ισότητες:

$$\sin(A-B) = \sin A \cos B - \cos A \sin B$$

$$\cos(A+B) = \cos A \cos B - \sin A \sin B$$

$$\cos(A-B) = \cos A \cos B + \sin A \sin B$$

$$\sin(2A) = 2 \sin A \cos A$$

$$\cos(2A) = \cos^2 A - \sin^2 A$$

$$\tan(2A) = (2 \tan A)/(1 - \tan^2 A)$$

$$\sin(3A) = 3 \sin A - 4 \sin^3 A$$

$$\cos(3A) = 4 \cos^3 A - 3 \cos A$$

$$\tan(3A) = (3 \tan A - \tan^3 A)/(1 - 3 \tan^2 A)$$

Σε κάθε πρόγραμμα να υπολογίσετε ξεχωριστά το κάθε μέλος των παραπάνω ισοτήτων και να το εκτυπώσετε για διάφορες τιμές των μεταβλητών A και B. Οι μεταβλητές A και B να δίνονται από την γραμμή εντολών. Δώστε μερικά παραδείγματα..

**2.2.3** Να γράψετε ένα πρόγραμμα το οποίο να δέχεται από το πληκτρολόγιο την θερμοκρασία σε βαθμούς Κελσίου και να την τυπώνει σε βαθμούς Fahrenheit. Δίνεται ότι  $F=1.8C+32$ .

**2.2.4** Να αναπτύξετε προγράμματα τα οποία να επαληθεύουν κάθε μία από τις ακόλουθες ισότητες

$$\log(AB) = \log A + \log B$$

$$\log(A/B) = \log A - \log B$$

$$\log(A^n) = n \log A$$

Σε κάθε πρόγραμμα να υπολογίσετε ξεχωριστά το κάθε μέλος των παραπάνω ισοτήτων και να το εκτυπώσετε για διάφορες τιμές των μεταβλητών A, B και n. Οι μεταβλητές A, B και n να δίνονται από το πληκτρολόγιο. Δώστε μερικά παραδείγματα.

**2.2.5** Να αναπτύξετε προγράμματα τα οποία να υπολογίζουν την αριθμητική τιμή για κάθε μία από τις ακόλουθες συναρτήσεις:

$$f(x) = \sqrt{\frac{5x^5 - 3x^3 + x - 1}{4}}$$

$$f(x) = \ln(x) * \sqrt{\frac{6x^2 - 4x + 4}{2 \ln(x)}}$$

$$f(x) = \ln(2x) * \sqrt{\frac{e^{3x} - 4}{5}}$$

και για διάφορες τιμές του x. Οι τιμές του x να εισάγονται στο πρόγραμμα από το πληκτρολόγιο. Εκτελέστε το πρόγραμμα δίνοντας τέσσερα δικά σας παραδείγματα.