



JAVA.

AN INTRODUCTION
TO OOP

"HELLO WORLD!"

Δημητριάδης Βασίλειος Κούσκουρας Ταξιάρχης

4 Διαλέξεις

Τρίτη – 8.00 μμ

Επόμενη Διάλεξη: **16/04/19**

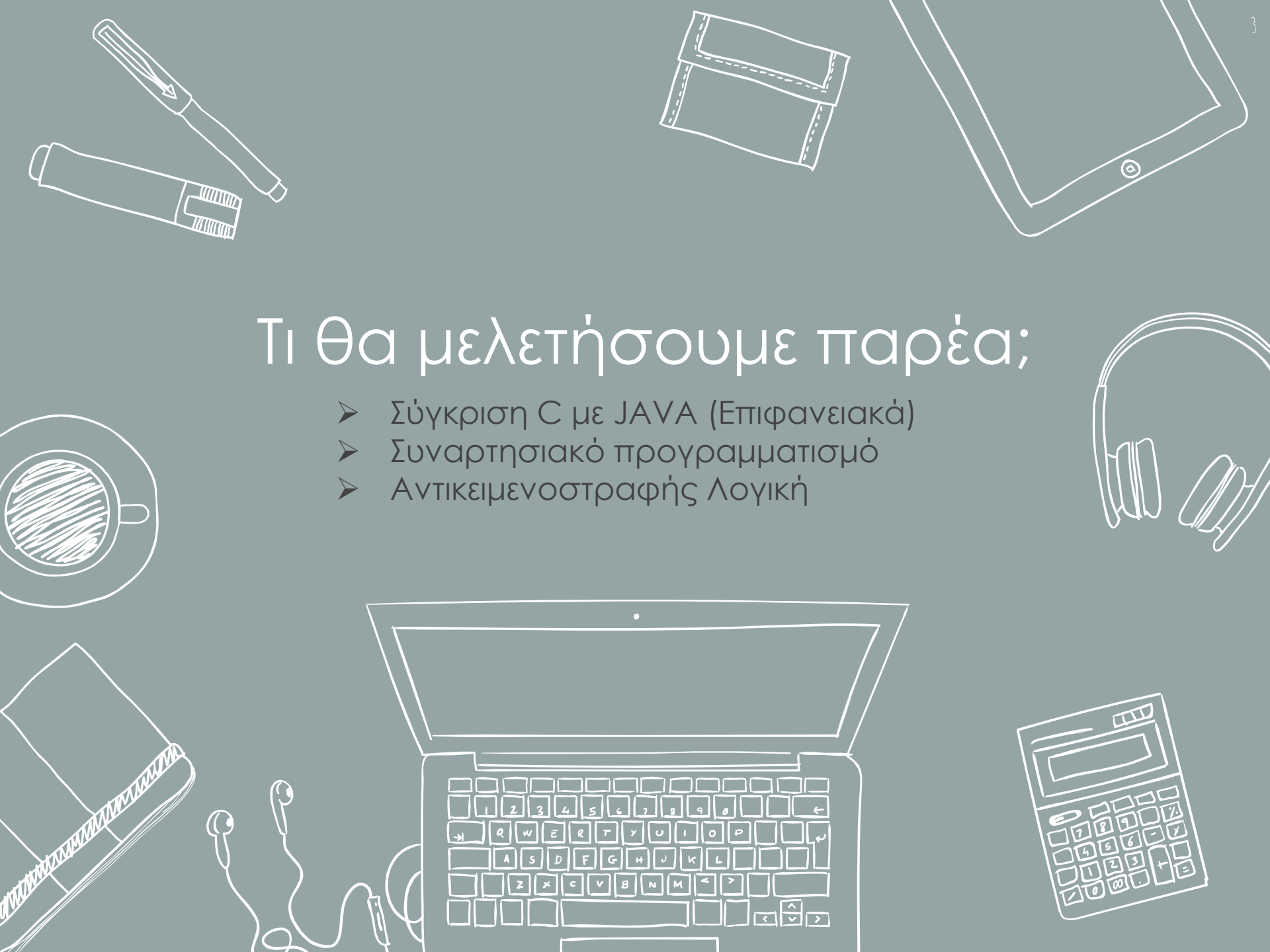
Μαζί με:

Κρανά Γεώργιο

Καράμπελα Άρη

Τι θα μελετήσουμε παρέα;

- Σύγκριση C με JAVA (Επιφανειακά)
- Συναρτησιακό προγραμματισμό
- Αντικειμενοστραφής Λογική



Γιατί Αντικειμενοστραφής Προγραμματισμός;

- ✓ Ευκολότερος στην ανάγνωση
- ✓ Επαναχρησιμοποίηση κώδικα
- ✓ Γλυτώνει χρόνο σε μεγάλα Project
- ✓ Ευκολότερος έλεγχος για λάθη
- ✓ Συντήρηση του λογισμικού
- ✓ Είναι πιο φυσικό



Το πρώτο μας πρόγραμμα

Με C:

```
#include <stdio.h>
int main(int argc, char *argv[])
{
    printf("Hello world!\n");
    return 0;
}
```

Η main του προγράμματος
μας

Public class πρέπει να έχει το ίδιο όνομα
με το αρχείο

Με Java:

```
public class HelloWorld {
    public static void main(String[] args) {
        // Prints "Hello, World" to the terminal window.
        System.out.println("Hello, World");
    }
}
```

Οι τρεις ροές δεδομένων

- Standard in: Stream-based input (**stdin**)
- Standard out: Stream-based output (**stdout**)
- Standard error: Stream-based error output (**stderr**)

```
public class HelloWorld {  
  
    public static void main(String[] args) {  
        // Prints "Hello, World" to the terminal window.  
        System.out.println("Hello, World");  
    }  
}
```

Λέξεις κλειδιά στην Java

abstract	assert	boolean	break
byte	case	catch	char
class	const	continue	default
do	double	else	enum
extends	final	finally	float
for	goto	if	implements
import	instanceof	int	interface
long	native	new	package
private	protected	public	return
short	static	strictfp	super
switch	synchronized	this	throw
throws	transient	try	void
volatile	while		

- ✓ Ποιες από αυτές ξέρετε;
- ✓ Τι σημαίνει αυτό για εμάς;



Μεταβλητές

Αποθήκευση και επεξεργασία τιμών

Βασικοί τύπο μεταβλητών στην JAVA

Type	Description	Default	Size	Example Literals
boolean	true or false	false	1 bit	true, false
byte	twos complement integer	0	8 bits	(none)
char	Unicode character	\u0000	16 bits	'a', '\u0041', '\\101', '\\\\', '\\\''', '\\n', 'β'
short	twos complement integer	0	16 bits	(none)
int	twos complement integer	0	32 bits	-2, -1, 0, 1, 2
long	twos complement integer	0	64 bits	-2L, -1L, 0L, 1L, 2L
float	IEEE 754 floating point	0.0	32 bits	1.23e100f, -1.23e-100f, .3f, 3.14F
double	IEEE 754 floating point	0.0	64 bits	1.23456e300d, -1.23456e-300d, 1e1d

Βασικές “ιδιότητες” μεταβλητών

Ορισμός Μεταβλητής

```
int myLuckyNumber;  
String myAddress;  
Boolean isAvailable;
```

Ανάθεση Μεταβλητής

```
int myLuckyNumber = 42;  
String myAddress = “21st Street”;  
Boolean isAvailable = false;
```

Ποιες είναι η **επιτρεπτές ονομασίες** όμως;

- ✓ Ξεκινάμε με γράμμα (A-Z) ή τα σύμβολα ‘_’ ή ‘\$’
- ✓ Έπειτα οποιοδήποτε αλφαριθμητικό
- ✓ Οι μεταβλητές είναι **case-sensitive**
- ✗ ΔΕΝ χρησιμοποιούμε λέξεις κλειδιά
- ✗ ΔΕΝ ξεκινάμε με αριθμό

Πως εκτυπώνω την τιμή μιας μεταβλητής στην Java;

Προσοχή στα “Class Types”

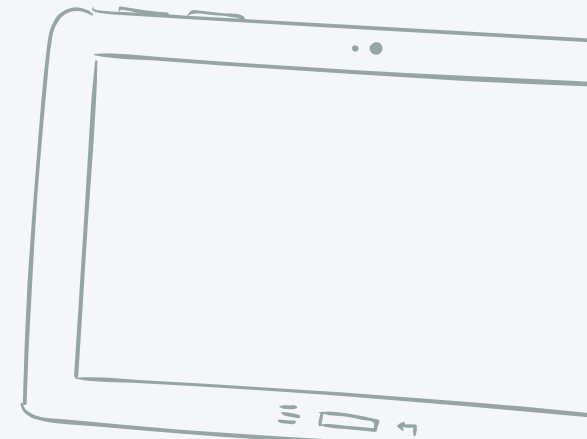
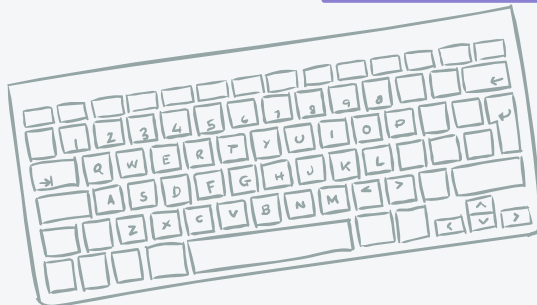
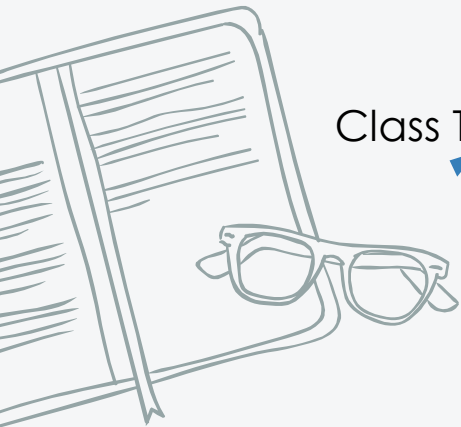
- Διαφορετικός τύπος δεδομένων
- Παρόμοιος με τους πρωταρχικούς, αλλά ξεκινάν με **κεφαλαίο** γράμμα
- Περιέχουν **μέσα έξτρα δυνατότητες**

Παράδειγμα

Class Type

```
Boolean isCorrect;  
boolean isAwesome;  
Integer myNumber;  
int anotherNumber;
```

Πρωταρχικός





Πράξεις

Διαθέσιμες πράξεις μεταξύ μεταβλητών

Αριθμητικοί Τελεστές

Τελεστές	Πράξη	Παραδείγματα
+	Πρόσθεση	$y = a + b$
-	Αφαίρεση	$y = a - b$
*	Πολλαπλασιασμός	$y = a * b$
/	Διαίρεση	$y = a / b$
%	Υπόλοιπο ακέραιας διαίρεσης	$y = a \% b$
++ και --	Αύξηση/Μείωση κατά 1	$y = a--; ++b;$

Σχεσιακοί Τελεστές

Τελεστές	Σχέση	Παραδείγματα (a = 12, b = 13)
==	Ισότητα	a == b : False
!=	Ανισότητα	a != b : True
>	Μεγαλύτερο	a > b: False
<	Μικρότερο	a < b: True
>=	Μεγαλύτερο ή Ίσο	a >= b: False
<=	Μικρότερο ή Ίσο	a <= b: True

Λογικοί Τελεστές

Τελεστές	Σχέση	Παραδείγματα (a = 12, b = 13)
&&	και	(a == b) && (a < b) : False
	ή	(a == b) (a < b) : True
!	όχι	!(a < b): False

Δυαδικοί Τελεστές

Τελεστές	Σχέση
&	Λογικό 'και'
	Λογικό 'ή'
^	Λογικό xor

Τελεστές	Σχέση
~	Συμπλήρωμα
<< ή >>	Λογικό shift
>>>	Zero-fill shift

Type-cast

- Μετατροπή μεταβλητής από τύπο σε τύπο
- Μεταξύ **πρωταρχικών τύπων** μόνο!
- Αυτόματα (*Widening*): Από μικρότερο σε μεγαλύτερο
- Χειροκίνητα (*Explicit*): Από μεγαλύτερο σε μικρότερο

Byte <-> Short <-> Int <-> Long <-> Float <-> Double

Automatic

```
int i = 100;  
long l = i;  
float f = l;
```

Explicit

```
double d = 100.04;  
long l = (long)d;  
int i = (int)l;
```




Δομές Επιλογής & Επανάληψης

Δίνοντας ζωή στο πρόγραμμα μας



Δομές ελέγχου

Έστω η ψευδοεντολή **print** για να γλυτώνουμε χώρο..

Δομή if

```
if(grade >= 5){  
    // Πάρτυ!  
}
```

Δομή if-else

```
if(grade >= 5) print("WOHOO");  
else print("\_(ツ)_/");
```

Χρησιμοποιώ το **one-line statement!**

Δομή if-else if-else

```
if(grade >= 8) print("WOHOO");  
else if(grade >= 5) print("Yeah!");  
else print("\_(ツ)_/");
```

Nesting

```
if(grade >= 8){  
    if(is_awesome){  
        print("Buy a Crepe");  
    }else print("Still WOHOO");  
}
```

Πότε χρησιμοποιώ Nesting;

Δομές ελέγχου

Δομή switch

```
switch(grade){  
  case 'A':  
    print("Excellent!");  
    break;  
  case 'B':  
  case 'C':  
    print("Well done");  
    break;  
  case 'D':  
    print("You passed");  
  case 'F':  
    print("Better try again");  
    break;  
  default :  
    print("Invalid grade");  
}
```

Προσοχή εδώ!
Τι γίνεται;

Τελεστής "? :" (Ternary)

```
print(x > y ? "x is greater" : "y is  
greater");
```

Ο Ternary operator χρησιμοποιείται συνήθως με τις **εντολές εκτύπωσης** ως συντόμευση Σε αυτά που θέλουμε να γράψουμε.

Πολλές φορές μπορεί να χρησιμοποιηθεί και σε συνεργασία με τα **return σε functions**.

Δομές επανάληψης

For-loop

```
for(int i=0;i<10;i++){  
    print("i == " + i);  
}
```

While-Loop

```
while(true){  
    print("Infinite spam");  
}
```

Do-While-Loop

```
int i = 0;  
do{  
    print("Infinite spam");  
    i++;  
}while(i <= -9000)
```

For-Each-Loop

```
String[] items = {"Sword",  
"Cupcake"};  
for(String item : items){  
    print("I have a: " + item);  
}
```

Μη χρήσιμο παράδειγμα.
Απλά δείχνουμε το πρόβλημα
του **infinite loop**!

Πότε χρησιμοποιώ ένα For-Each-Loop;

Έλεγχος επαναληπτικών δομών

Εντολή Break

```
int[] nums = {10, 20, 50};  
for(int num : nums){  
    if(num == 20) break;  
    print(num);  
}
```

Εντολή Continue

```
int[] nums = {3, 5, 12, 30, 8};  
for(int num : nums){  
    if(num < 10) continue;  
    print(num);  
}
```



Είσοδος από πληκτρολόγιο

Διαβάζοντας δεδομένα από τους χρήστες μας



Πως γίνεται να δεχτούμε user-input;

```
package helloworld;  
import java.util.Scanner;
```

Πρέπει να κάνουμε
import το Scanner
class

```
/**  
 *  
 *  
 */
```

```
public class HelloWorld {
```

Φτιάχνουμε ένα Scanner
object

```
/**  
 * @param args the command line arguments  
 */  
public static void main(String[] args) {  
    // TODO code application logic here  
    Scanner scan = new Scanner(System.in);
```

Τι κάνει αυτή η λέξη;

Ποή Standard Input (**stdin**)

Ας δούμε ένα πραγματικό παράδειγμα

```
public class HelloWorld {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        // TODO code application logic here  
        Scanner scan = new Scanner(System.in);  
  
        System.out.println("Type your Name: ");  
        String name = scan.nextLine();  
        System.out.println("Type your age: ");  
        int age = scan.nextInt();  
        System.out.println("Type your height: ");  
        float height = scan.nextFloat();  
  
        System.out.println(name);  
        System.out.println(age);  
        System.out.println(height);  
    }  
}
```

Μέθοδοι για
διαφορετικούς
τύπους
ματαβλητών

Που μπορούμε να δούμε αναλυτικά τον Scanner;

- Η κλάση *Scanner* έχει **40+ μεθόδους**
- Πρέπει να τις ξέρουμε όλες; Πως θα τις θυμόμαστε;
- Σημαντικές μέθοδοι:
next**Line**(), next**Int**(), next**Short**() κλπ..

<https://docs.oracle.com/javase/7/docs/api/java/util/Scanner.html>

'Ελεγχος εισόδου

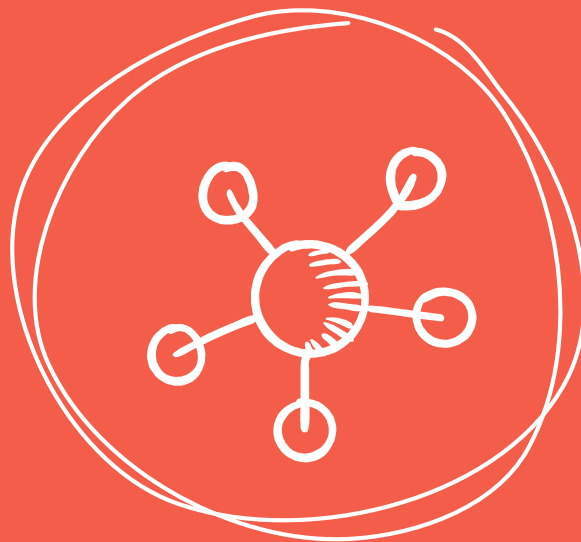
```
public static void main(String[] args) {  
    // TODO code application logic here  
    Scanner scan = new Scanner(System.in);  
    int number;
```

```
do{  
    System.out.println("Enter a positive int");  
    while(!scan.hasNextInt()){  
        System.out.println("Thats not a int");  
        scan.next();  
    }  
    number = scan.nextInt();  
}while(number < 0);
```

```
System.out.println(number);  
}
```

Είναι ο αριθμός
μεγαλύτερος από
0;

Είναι η είσοδος
αριθμός;



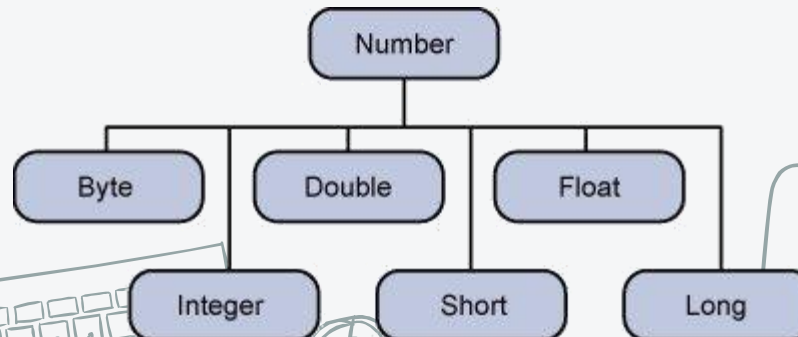
Βοηθητικές Κλάσεις

Αντικείμενα για την επεξεργασία αριθμών και
αλφαριθμητικών



Η κλάση “Number”

- Όλες οι *Class Types* είναι παιδιά μίας ομάδας γνωστή και ως *Number*.
- Τα παιδιά αυτής της κλάσης ονομάζονται “υπό-κλάσεις”
- Η κλάση *Number* περιέχει μέσα βοηθητικές συναρτήσεις όπως μετατροπή αριθμών από ένα τύπο σε άλλο.
- Βέβαια, σπάνια χρησιμοποιείται άμεσα.



Η κλάση “Math”

- Περιέχει μέσα βασικές αριθμητικές συναρτήσεις (54).
- Περιέχει μέσα τον αριθμό π και e .
- Χρησιμοποιείται αρκετά συχνά στα προγράμματα μας
- Βέβαια, σπάνια χρησιμοποιείται άμεσα.

<https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html>

Method name	Description
<code>Math.abs (value)</code>	absolute value
<code>Math.ceil (value)</code>	rounds up
<code>Math.floor (value)</code>	rounds down
<code>Math.log10 (value)</code>	logarithm, base 10
<code>Math.max (value1, value2)</code>	larger of two values
<code>Math.min (value1, value2)</code>	smaller of two values
<code>Math.pow (base, exp)</code>	<i>base</i> to the <i>exp</i> power
<code>Math.random ()</code>	random double between 0 and 1
<code>Math.round (value)</code>	nearest whole number
<code>Math.sqrt (value)</code>	square root
<code>Math.sin (value)</code>	sine/cosine/tangent of an angle in radians
<code>Math.cos (value)</code>	
<code>Math.tan (value)</code>	
<code>Math.toDegrees (value)</code>	convert degrees to radians and back
<code>Math.toRadians (value)</code>	

Constant	Description
<code>Math.E</code>	2.7182818...
<code>Math.PI</code>	3.1415926...

Η κλάση “String”

- Η πιο συνηθισμένη βοηθητική κλάση.
- Πληθώρα βοηθητικών συναρτήσεων (40+).

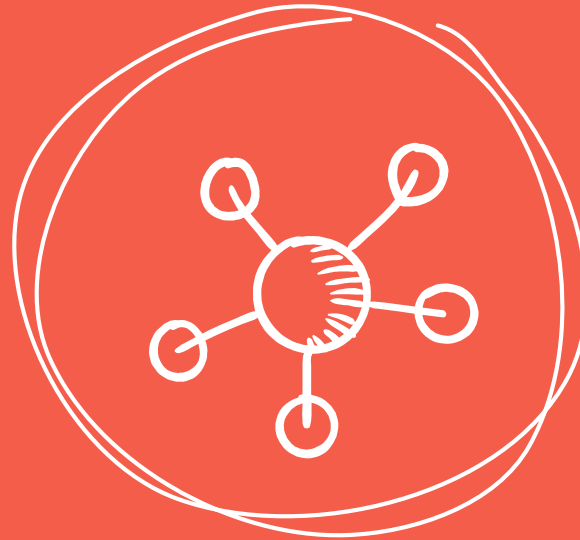
<https://docs.oracle.com/javase/8/docs/api/java/lang/String.html>

Java String Methods

(1) length()	(8) toUpperCase()	(15) compareTo
(2) charAt()	(9) split()	(16) startsWith()
(3) trim()	(10) substring()	(17) endsWith()
(4) indexOf()	(11) equals()	⋮
(5) lastIndexOf()	(12) getBytes()	⋮
(6) replace()	(13) concat()	etc.
(7) toLowerCase()	(14) contains()	

Παράδειγμα χρήσης "String"

```
String addr1 = "123 Baker's Street";  
String addr2 = "123 Baker's STREET";  
print(addr1 == addr2); false  
print(addr1.equals(addr2)); false  
print(addr1.equalsIgnoreCase(addr2)); true  
print(addr1.compareTo(addr2)); 32  
print(addr1.compareToIgnoreCase(addr2)); 0  
print(addr1.toUpperCase()); 123 BAKER'S STREET  
print(addr1.startsWith("Bakers")); false  
print(addr1.contains("Baker's")); true  
print(addr1.substring(0, 11)); 123 Baker's
```



Στατικοί Πίνακες

Συλλογή πολλών όμοιων αντικειμένων σε έναν πίνακα.



Ορισμός Στατικού Πίνακα

```
dataType[] arrayVar;  
dataType arrayVar[];
```

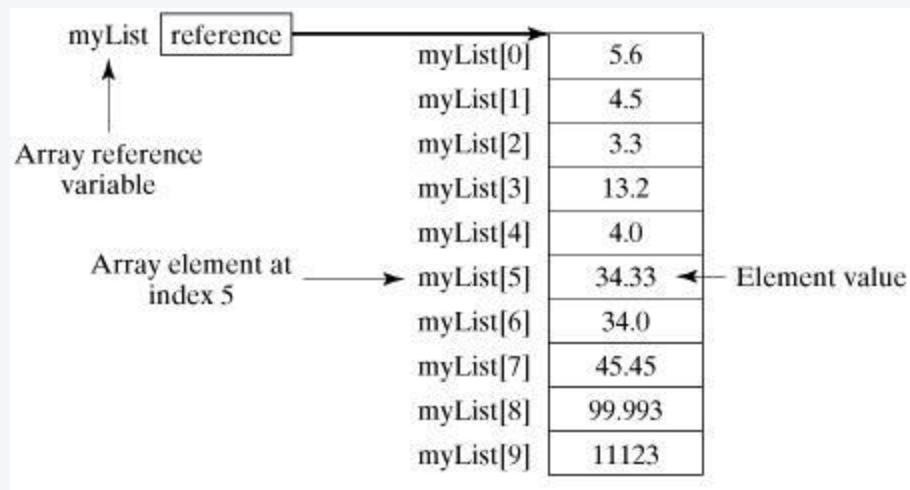


```
double[] myList;  
double myList[];
```

Δημιουργία Στατικού Πίνακα

```
Weapon[] myWeapons = new Weapon[3];  
Cupcake[] myCC = {Cupcake1, Cupcake2};  
Double[] myList = {5.6, 4.5, 3.3, 13.2, 4.0};
```

Τελικά τι φτιάχνεται από αυτό;



Πως επεξεργαζόμαστε στοιχεία;

```
double[] myList = {1.9, 2.9, 3.4, 3.5};
```

Εμφάνιση όλων

```
for (int i = 0; i < myList.length; i++) {  
    print(myList[i] + " ");  
}
```

Υπολογισμός Συνόλου

```
double total = 0;  
for (int i = 0; i < myList.length; i++) {  
    total += myList[i];  
}
```

Υπολογισμός Μεγίστου

```
double max = myList[0];  
for (int i = 1; i < myList.length; i++) {  
    if (myList[i] > max) max = myList[i];  
}
```

Και ένα πλήρες παράδειγμα με *for-each*

```
public class TestArray {  
  
    public static void main(String[] args) {  
        double[] myList = {1.9, 2.9, 3.4, 3.5};  
  
        for (double element: myList) {  
            System.out.println(element);  
        }  
    }  
}
```

Η κλάση "Arrays"

- 4^{ης} Βασικές συναρτήσεις
- `binarySearch()`, `equals()`, **`fill()`**, **`sort()`**

Παράδειγμα Fill

```
int ar[] = {2, 2, 1, 8, 3, 2, 2, 4, 2};  
Arrays.fill(ar, 10);
```



```
Array completely filled with 10  
[10, 10, 10, 10, 10, 10, 10, 10, 10]
```

Παράδειγμα Sort

```
int[] arr = {13, 7, 6, 45, 21, 9, 101, 102};  
Arrays.sort(arr);
```



```
Modified arr[] : [6, 7, 9, 13, 21, 45, 101, 102]
```



Το Netbeans και τα Πακέτα

Πως μοιάζει το Netbeans και τι είναι τα πακέτα στην Java; Που και πως τα χρησιμοποιούμε;



Netbeans IDE

TOOLBAR

Project Window
File-Structure Tree

Code

Navigator Tree

Output / Console

Java Packages

- Χρησιμοποιούνται για να μην γίνεται **σύγκρουση ονομάτων**.
- Βοηθούν στην **ομαδοποίηση του κώδικα**.
- Γνωστά πακέτα: *java.lang*, *java.io*
- Πολύ συχνά φτιάχνουμε δικά μας πακέτα, για να ξεχωρίζουν τα προγράμματα μας από προγράμματα άλλων.
- Μπορούμε να τα σκεφτούμε και ως «φακέλους».

Κακιά ονομασία

```
package Fireworks;
```

```
import java.util.ArrayList;
import java.util.Iterator;
.....
public class Fireworks extends Application {
    .....
}
```

Καλή ονομασία

```
package com.wckdawe.Fireworks;
```

```
import java.util.ArrayList;
import java.util.Iterator;
.....
public class Fireworks extends Application {
    .....
}
```


Κοινό package

Παραπάνω για τα Packages

Διαφορετικό package

Animal.java

package animals;

```
interface Animal {
    public void eat();
    public void travel();
}
```

Mammal.java

package animals;

```
public class Mammal implements Animal {

    public void eat() {
        System.out.println("Mammal eats");
    }

    public void travel() {
        System.out.println("Mammal travels");
    }

    public int noOfLegs() {
        return 0;
    }
}
```

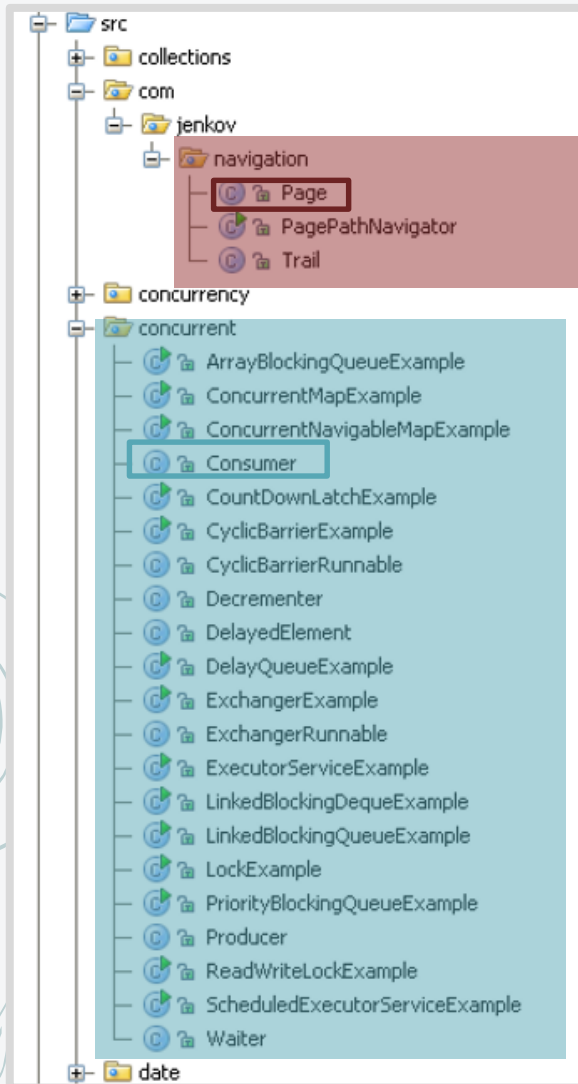
MyApp.java

package com.wckdawe.MyApp;**import animals.Mammal;**

```
public class MyApp{
    public static void main(String args[]) {
        Mammal m = new Mammal();
        m.eat();
        m.travel();
    }
}
```

- Τα κοινά πακέτα υποδηλώνουν ότι προέρχονται από την ίδια εφαρμογή.
- Όταν ήμαστε σε διαφορετικό πακέτο, πρέπει να κάνουμε `import` το πακέτο του «τρίτου».
- Ουσιαστικά, κάνουμε **import** (include) κομμάτια άλλων βιβλιοθηκών.

Η λέξη κλειδί `import`



Υπάρχουν δύο τρόποι για να κάνουμε `include`:

- Ακριβής: `import com.jenkov.navigation.Page;`
- Γενικός: `import com.jenkov.navigation.*;`

Ο **αστερίσκος** δηλώνει ότι συμπεριλαμβάνουμε **κάθε** κλάση που είναι μέσα σε εκείνο το πακέτο.

Αυτά από εμένα.

Δημητριάδης Βασίλειος

The answer to life, the universe, and everything