

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό – Διάλεξη #4

Διάλεξη #4:

Τύποι δεδομένων και τελεστές, εντολές επιλογής


Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 1

Απλοί τύποι σε αντιπαράθεση με αντικείμενα

- Η Java παρέχει τύπους αντικειμένων και τύπους βασικών δεδομένων (primitive data types)

Οι βασικοί τύποι δεδομένων είναι: Τύποι αντικειμένων:

Όλοι οι άλλοι!
(περιλαμβανομένων των Strings!)



Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 2

Βασικοί τύποι δεδομένων

τύπος	μέγεθος (σε bit)	ενδεικτικές τιμές
Ακέραιοι (integral numbers):		
byte	8	28
short	16	28
int	32	28, 034, 0x1C
long	64	28L
Αριθμοί κινητής υποδιαστολής (floating point numbers):		
float	32	1.234f
double	64	1.234, 1.34e3
Λογικές τιμές (boolean values):		
boolean	8	true, false
Χαρακτήρες (characters):		
char	16	'c', '\u4567'

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 3

Τύποι ακεραίων αριθμών

```
int age = 35;
short shoeSize = 8;

int a;
long b;
b = a; // ok: coercion (widening conversion)

«Μικρότεροι» αριθμητικοί τύποι δεδομένων μπορούν να καταχωρηθούν σε μεγαλύτερους.

int a;
char c;
a = c; // ok!

Οι χαρακτήρες θεωρούνται αριθμητικά δεδομένα! Έχουν αριθμητική τιμή και μπορεί να καταχωρηθούν σε ακέραιες μεταβλητές.
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 4

Τύποι κινητής υποδιαστολής (Floating point)

```
float f = 3.1415f;
double x = 42.0;

int i;
float f;
f = i; // ok: coercion (widening conversion)

int i;
float f;
f = f + i; // ok: i will be promoted to float
i = i + f; // error: f will not be narrowed
// (automatically) to an int
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 5

Λογικές τιμές (Boolean)

```
boolean done = false;
boolean isWeekend = true;

isWeekend = (dayIndex == 6) || (dayIndex == 7);

(Η ερμηνεία της έκφρασης θα δοθεί σε λίγο...)
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 6

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό – Διάλεξη #4

Ο τύπος χαρακτήρων

```
char ch;  
ch = 'm';           // note: single quotes!  
  
ch = '\u2456';     // unicode  
ch = '\u0008';     // backspace; unicode is superset  
                  //of ASCII  
  
int code = ch;     // this is legal!
```

Οι χαρακτήρες αποθηκεύονται μέσω του αριθμητικού τους κώδικα. Αποθηκεύονται όπως οι ακέραιοι αριθμοί και μπορεί να χρησιμοποιηθούν (σχεδόν) σαν αριθμοί.

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 7

Τελεστές (Operators)

Ενός έντελου (unary)	++ -- + - ! ~ ()
Αριθμητικοί	* / % + -
Ολίσθησης (shift)	<< >> >>>
Συγκρίσης (comparisons)	< <= > >= == !=
Bit προς bit (bitwise binary)	& ^
Λογικοί (logical)	&& (short-circuit)
Τριαδικός (ternary)	?:
Καταχώρησης (assignment)	= op=

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 8

Τελεστές σε επίπεδο bit

!	Αντιστροφή, άρνηση
~	bit προς bit αντιστροφή
&	bitwise AND
	bitwise OR
^	bitwise XOR

Δεν θα ασχοληθούμε άλλο με τους τελεστές σε επίπεδο bit

<<	ολίσθηση προς τα αριστερά
>>	ολίσθηση προς τα δεξιά
>>>	ολίσθηση προς τα δεξιά χωρίς επέκταση πρόσημου

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 9

Αριθμητικοί τελεστές

+	πρόσθεση
-	αφαίρεση
*	πολλαπλασιασμός
/	διαίρεση
%	υπόλοιπο

```
a = b + 32;  
n = (a + 1) * (b + 1);  
  
share = amount / people;  
remains = amount % people;
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 10

Αύξηση / μείωση μίας μονάδας

n++	εκ των υστέρων αύξηση
n--	εκ των υστέρων μείωση
++n	εκ των προτέρων αύξηση
--n	εκ των προτέρων μείωση

```
x++;           // increment x by one  
x--;           // decrement x by one  
y = 8;  
x = y++;  
x = ++y;
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 11

Τελεστές σύγκρισης

>	Μεγαλύτερο από
<	Μικρότερο από
>=	Μεγαλύτερο από ή ίσο με
<=	Μικρότερο από ή ίσο με
==	Ίσο με
!=	Όχι ίσο με

Οι τελεστές σύγκρισης επιστρέφουν μια λογική (boolean) τιμή.

```
boolean isHD = (mark >= 80);  
  
return (number == 0);
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 12

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό – Διάλεξη #4

Λήψη αποφάσεων

Παράδειγμα:

```
System.out.println("Your mark is: " + mark);  
  
if (mark > 70)  
    System.out.println("well done!");  
  
System.out.println("Talk to your tutor for details");
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 13

Λήψη αποφάσεων (2)

Άλλο ένα παράδειγμα:

```
System.out.println("Your mark is: " + mark);  
  
if (mark >= 50)  
    System.out.println("You have passed.");  
else  
    System.out.println("Please try again.");  
  
System.out.println("Talk to your tutor for details");
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 14

Η εντολή επιλογής if

```
if (συνθήκη)  
    εντολή;  
else  
    εντολή;
```

```
if (συνθήκη)  
{  
    εντολή;  
    εντολή;  
    ...  
}  
else  
{  
    εντολή;  
    εντολή;  
    ...  
}
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 15

Τμήματα κώδικα (blocks)

Τα άγκιστρα { } ομαδοποιούν το σύνολο των εντολών που περικλείουν. Το σύνολο των εντολών αυτών καλείται «τμήμα κώδικα». Τα τμήματα κώδικα αντιμετωπίζονται σαν μια απλή εντολή από τις δομές έλεγχου ροής.

```
{  
    εντολή;  
    εντολή;  
    εντολή;  
    ...  
}
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 16

Παράδειγματα

```
if (value >= 1 && value <= 10)  
{  
    ...  
}  
else  
{  
    ...  
}
```

```
if ((value >= 1) && (value <= 10))  
{  
    ...  
}
```

```
if (processFinished == true) ← άσχημο  
{  
    ...  
}  
if (processFinished) ← καλύτερο  
{  
    ...  
}
```

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 17

Τιμές βασικών τύπων και αναφορές

- Τα δεδομένα βασικών τύπων αποθηκεύονται **κατ' αξία** [by value], ενώ τα αντικείμενα αποθηκεύονται **κατ' αναφορά** [by reference]

age = 42;

42

father = new Person();

Person

name = "Fred";

"Fred"

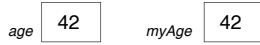
Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 18

Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό – Διάλεξη #4

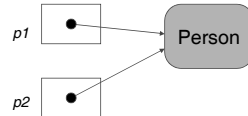
Η εντολή καταχώρησης (για 2^η φορά)

- Η καταχώρηση γίνεται κατ' αξία για τους βασικούς τύπους δεδομένων και κατ' αναφορά για τύπους αντικειμένων

```
age = 42;  
myAge = age;
```



```
p1 = new Person();  
p2 = p1;
```



Εισαγωγή στον Αντικειμενοστρεφή Προγραμματισμό, Αντώνιος Συμβώνης, ΣΕΜΦΕ, ΕΜΠ, Slide 19