



Τύποι Δεδομένων (Java Type System)

Αθ. Ανδρούτσος



Προγράμματα και Δεδομένα

Προγραμματισμός με Java

- Ένα πρόγραμμα είναι μία **ακολουθία εντολών**
- Οι εντολές επεξεργάζονται **δεδομένα (data)**
- Τα δεδομένα μπορεί να είναι αριθμοί, χαρακτήρες, κλπ. αυτός είναι ο **τύπος τους (data type)**
- Υπάρχουν **πρωταρχικοί (primitives)** τύποι δεδομένων και **σύνθετοι τύποι** δεδομένων



Τύποι Δεδομένων

Προγραμματισμός με Java

- Δύο ειδών τύποι Δεδομένων:
 1. **Πρωταρχικοί Τύποι Δεδομένων** (Primitive Types) ή απλοί τύποι δεδομένων όπως αριθμοί, χαρακτήρες, τιμές αλήθειας
 2. **Σύνθετοι Τύποι Δεδομένων** (Reference Types) συνθέτονται από απλούς ή άλλους σύνθετους τύπους



Πρωταρχικοί Τύποι Δεδομένων

Προγραμματισμός με Java

- Πρωταρχικοί τύποι (Primitives)
 - **Αριθμοί**, όπως ακέραιοι, δεκαδικοί
 - **Χαρακτήρες**, όπως, π.χ. οι εκτυπώσιμοι χαρακτήρες
 - **Τιμές αλήθειας** (true, false)



Δεδομένα και μεταβλητές

Προγραμματισμός με Java

- Τα δεδομένα ενός προγράμματος αποθηκεύονται, όπως και οι εντολές, στην κύρια μνήμη (ή απλά μνήμη)
- Τα δεδομένα **αποθηκεύονται σε μεταβλητές**, οι οποίες έχουν ένα όνομα ή αναγνωριστικό (identifier) και ανήκουν σε ένα τύπο δεδομένων
- Σε γλώσσες strongly-typed όπως η Java **οι μεταβλητές πρέπει να ανήκουν σε ένα τύπο δεδομένων**



Τύποι Δεδομένων και Java

Προγραμματισμός με Java

- Με βάση τον τύπο δεδομένων ορίζονται:
 - Εύρος τιμών που μπορούν να αποδοθούν στα δεδομένα
 - Πράξεις που μπορούν να εκτελεστούν στις τιμές αυτές
- Η Java παρέχει (i) **πέντε πρωταρχικούς τύπους δεδομένων** (οκτώ μαζί με τρεις υποτύπους που θα δούμε) καθώς και (ii) **σύνθετους τύπους δεδομένων**



Πρωταρχικοί Τύποι Δεδομένων (1)

Προγραμματισμός με Java

- ***int*** - Ακέρατοι με μέγεθος 4 bytes (32-bits)
 - byte - Ακέρατοι με μέγεθος 1 byte (8-bits)
 - short - Ακέρατοι με μέγεθος 2 bytes (16-bits)
 - long - Ακέρατοι με μέγεθος 8 bytes (64-bits)
 - ***char*** 2 bytes (16-bits) - Εκτυπώσιμοι χαρακτήρες
 - ***boolean*** 1 byte (8-bits) – Αλήθεια/Ψέμα
- Αυτοί οι τύποι δεδομένων είναι **ακέρατοι** με διαφορετικό μέγεθος από τον τύπο `int`



Πρωταρχικοί Τύποι Δεδομένων (2)

Προγραμματισμός με Java

- ***float*** 4 bytes (32-bits)
 - Δεκαδικοί Αριθμοί με υποδιαστολή
- ***double*** 8 bytes (64-bits)
 - Δεκαδικοί Αριθμοί με περισσότερα δεκαδικά ψηφία



Πρωταρχικοί Τύποι Δεδομένων (3)

Προγραμματισμός με Java

type:byte	size:8	min:-128	max:127
type:short	size:16	min:-32768	max:32767
type:char	size:16	min:0	max:65535
type:int	size:32	min:-2147483648	max:2147483647
type:long	size:64	min:-9223372036854775808	max:9223372036854775807
type:float	size:32	min:1.4E-45	max:3.4028235E38
type:double	size:64	min:4.9E-324	max:1.7976931348623157E308

- Αν αποθηκεύσουμε σε μία μεταβλητή ενός τύπου δεδομένων τιμή μεγαλύτερη ή μικρότερη από τις MIN/MAX τιμές θα δημιουργηθεί υπερχείλιση (overflow) και το αποτέλεσμα δεν θα είναι ορθό



Δηλώσεις Μεταβλητών

Προγραμματισμός με Java

- Οι μεταβλητές ενός προγράμματος θα πρέπει **πριν χρησιμοποιηθούν να δηλώνονται** μέσα στο πρόγραμμα ώστε ο μεταγλωττιστής να καταχωρεί μνήμη
- Η **δήλωση** (declaration) μιας μεταβλητής (variable declaration) περιλαμβάνει τον **τύπο** και το **αναγνωριστικό της** (δηλ. το όνομά της)
 - Παράδειγμα:
 - `int booksNumber;`
 - `boolean flag;`



Τιμές μεταβλητών

Προγραμματισμός με Java

- Η τιμή μιας μεταβλητής είναι τα **περιεχόμενα της θέσης μνήμης** που αντιστοιχεί στη μεταβλητή αυτή
- Μπορούμε **ταυτόχρονα με τη δήλωση να αρχικοποιήσουμε μία μεταβλητή** π.χ.
`int sum = 30;` – αλλιώς τα περιεχόμενά της είναι απροσδιόριστα (ή στην περίπτωση μεταβλητών κλάσεων ορίζονται από τον μεταγλωττιστή)



Μεταβλητές απλών τύπων

Προγραμματισμός με Java

- Θέσεις μνήμης που περιέχουν την τιμή ενός συγκεκριμένου απλού τύπου δεδομένων

Μνήμη

result	30

`int result = 30;`

Με αυτή την εντολή ο μεταγλωττιστής της Java **δεσμεύει στη μνήμη** χώρο για έναν **ακέραιο** και ονομάζει αυτό τον χώρο **result**. Επίσης εκχωρεί την **τιμή 30**.

Ως χώρο που δεσμεύεται στη μνήμη **απεικονίζουμε συνήθως ένα κελί (byte)** της μνήμης, στην πραγματικότητα όμως δεσμεύονται περισσότερα bytes, ανάλογα με τον τύπο δεδομένων. Για παράδειγμα οι **ακέραιοι καταλαμβάνουν 4 bytes** στη μνήμη (32-bits)



Μνήμη και πρωταρχικοί τύποι

Προγραμματισμός με Java

- Η Java χωρίζει τη μνήμη σε δύο βασικά μέρη: το **Stack** (στοίβα) και το **Heap** (Σωρός) και χρησιμοποιεί το Stack για στατική δέσμευση μνήμης δηλαδή δέσμευση κατά το χρόνο μεταγλώττισης

Μνήμη Stack

Οι Πρωταρχικοί τύποι δεδομένων όπως ο τύπος Ακεραίων αποθηκεύονται πάντα στο Stack

result

30

```
int result = 30;
```

Κατά το **χρόνο μεταγλώττισης** δεσμεύεται χώρος στο Stack για την μεταβλητή result



Αναγνωριστικά (Identifiers)

Προγραμματισμός με Java

- Ονόματα που επιτρέπει η Java να δίνουμε σε μεταβλητές και άλλα αντικείμενα ενός προγράμματος
- Ξεκινούν με γράμμα, `_` , `$` (όχι με ψηφίο), και οι επόμενοι χαρακτήρες μπορούν να περιλαμβάνουν γράμματα, αριθμητικά ψηφία, `_` ή `$`



Αναγνωριστικά

Προγραμματισμός με Java

- Δεσμευμένες λέξεις (reserved words) όπως `int`, `char`, `class`, κλπ., απαγορεύεται να χρησιμοποιούνται ως αναγνωριστικά
- Τα πεζά γράμματα δεν είναι ίδια με κεφαλαία (Η Java είναι case sensitive)
 - Η μεταβλητή `num` δεν είναι το ίδιο με το `Num` ή `NUM` ή `nUM` ή `nuM`, κλπ.



Ονοματοδοσία αναγνωριστικών

Προγραμματισμός με Java

- Δηλώνουν τη σημασία του αντικειμένου που προσδιορίζουν
- Στις μεταβλητές κατά σύμβαση (*convention*) το πρώτο γράμμα είναι μικρό και το πρώτο γράμμα κάθε λέξης που περιέχεται κεφαλαίο (Camel Case)
- Στις κλάσεις κατά σύμβαση το πρώτο γράμμα είναι κεφαλαίο και το πρώτο γράμμα κάθε λέξης που περιέχεται επίσης κεφαλαίο (Pascal Case)



Αναγνωριστικά Μεταβλητών

Προγραμματισμός με Java

- Τα ονόματα που δίνουμε στις μεταβλητές (αναγνωριστικά - identifiers) είναι κατά σύμβαση το πρώτο γράμμα πεζό και στη συνέχεια το πρώτο γράμμα κάθε λέξης κεφαλαίο
- Παράδειγμα:

```
int booksNumber;  
boolean flag;
```



Σταθερές (1)

Προγραμματισμός με Java

- Μερικές μεταβλητές δεν θέλουμε να μεταβάλλουν την τιμή τους και μπορούν να παρασταθούν ως σταθερές με τη λέξη *final*
- Τα ονόματα σταθερών είναι κατά σύμβαση κεφαλαία
 - π.χ. `final double PI = 3.14;`



Σταθερές (2)

Προγραμματισμός με Java

- Οι σταθερές, όπως αναφέραμε, δηλώνονται με τη λέξη `final`
 - `final int ALICE_AGE = 22;`
 - `final boolean FLAG = false;`
- Στα αναγνωριστικά σταθερών κατά σύμβαση χρησιμοποιούμε κεφαλαία γράμματα και κάτω παύλα (`_`) μεταξύ των λέξεων



Σταθερές – Σφάλματα

Προγραμματισμός με Java

- Η προσπάθεια μεταβολής της τιμής μιας μεταβλητής `final` προκαλεί σφάλμα μεταγλώττισης



Εκχωρήσεις

Προγραμματισμός με Java

- Στις μεταβλητές μπορούμε να εκχωρούμε τιμές με τον τελεστή `=`. Το σύμβολο `=` δεν υποδηλώνει ισότητα αλλά είναι **σύμβολο εκχώρησης** (μπορεί να διαβαστεί και ως 'γίνεται')
- Μπορούμε να εκχωρούμε **κυριολεκτικές τιμές (literals)** ή **παραστάσεις (expressions)**
- Εκχώρηση literals (κυριολεκτικών τιμών)
 - `int age = 20;`
- Εκχώρηση παράστασης (expression)
 - `int nextAge = age + 1;`



Εκχώρηση κυριολεκτικών (literals)

Προγραμματισμός με Java

- Ως *‘κυριολεκτικά’* (*literals*) αναφερόμαστε στις τιμές αυτές καθαυτές
- Τα κυριολεκτικά (οι αριθμοί δηλαδή αυτοί καθαυτοί) αναπαρίστανται -όπως παρακάτω το 35- χωρίς μονά ή διπλά εισαγωγικά
 - `int sum = 35;`



Εκχώρηση παραστάσεων

Προγραμματισμός με Java

- `int sum = a + s + 12;`
- `int mul = sum * 7;`
- `int div = sum / 4;`
- `int mod = sum % 4;`

Σε όλες τις παραστάσεις (expressions) πρώτα υπολογίζεται το δεξί μέρος (Right-hand side) και στη συνέχεια εκχωρείται στην μεταβλητή αριστερά (Left-hand side)



Αρχικοποίηση

Προγραμματισμός με Java

- Τα περιεχόμενα των μεταβλητών αν δεν τα αρχικοποιήσουμε εμείς **έχουν απροσδιόριστες τιμές** και δημιουργούν σφάλματα μεταγλώττισης
- **Θα πρέπει να αρχικοποιούμε** εμείς τις μεταβλητές μας συνήθως κατά τη στιγμή της δήλωσης
- Εξαίρεση αποτελούν οι μεταβλητές κλάσεων που όπως θα δούμε αρχικοποιούνται αυτόματα από τον μεταγλωττιστή, ακόμα κι αν δεν τις αρχικοποιήσουμε εμείς



Παράδειγμα

Προγραμματισμός με Java

```
1 package gr.aueb.cf.ch1;
2
3 /**
4  * Προσθέτει δύο ακεραίους και εμφανίζει το
5  * αποτέλεσμα στην οθόνη (κονσόλα).
6  *
7  * @author a8ana
8  */
9 public class AddApp {
10
11     public static void main(String[] args) {
12         // Δήλωση και αρχικοποίηση μεταβλητών
13         int num1 = 5;
14         int num2 = 12;
15         int result = 0;
16
17         // Εντολές
18         result = num1 + num2;
19
20         // Εκτύπωση αποτελέσματος
21         System.out.println("Το αποτέλεσμα είναι: " + result);
22     }
23 }
```

- Θα πρέπει τώρα να είστε σε θέση να αντιληφθείτε το πρόγραμμα
- Το + μέσα στην println λειτουργεί ως **τελεστής συνένωσης** κειμένου και μεταβλητής (concat operator)



Αποτέλεσμα

Προγραμματισμός με Java

```
Run: AddApp x
"C:\Program Files\Amazon Corrn
Το αποτέλεσμα είναι: 17
```

- Το αποτέλεσμα της πρόσθεσης $5 + 12$ είναι 17, δηλαδή το αναμενόμενο



Ασκήσεις - Νέο Project

Προγραμματισμός με Java

New Project

Search

New Project

Empty Project

Generators

- Maven Archetype
- JavaFX
- Kotlin Multiplatform
- Compose Multiplatform
- IDE Plugin
- Android

Name: CodingFactoryTestbed

Location: ~\IdeaProjects\idea922

Project will be created in: ~\IdeaProjects\idea922\CodingFactoryTestbed

☐ Create Git repository

Language: Java Kotlin Groovy HTML +

Build system: IntelliJ Maven Gradle

JDK: 11 Amazon Corretto version 11.0.10

☐ Add sample code

> Advanced Settings

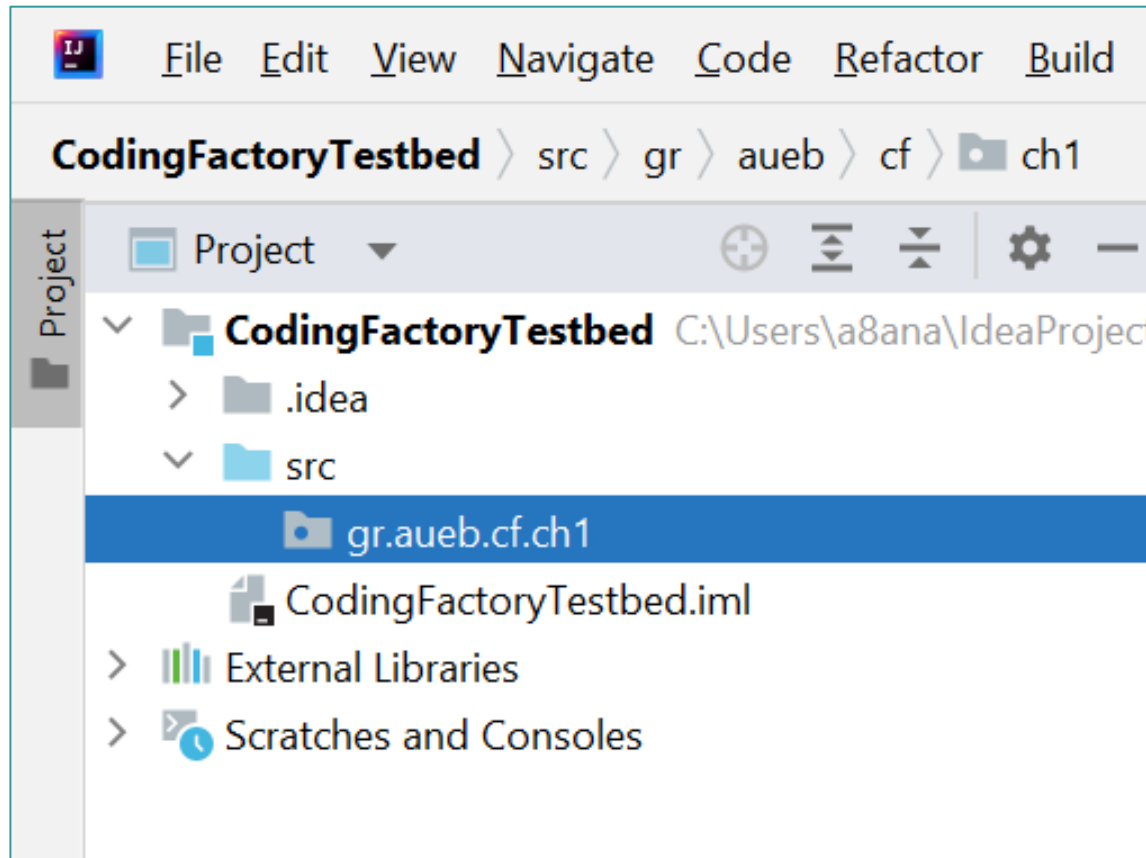
Create Cancel

- Δημιουργήστε ένα νέο Java project με όνομα ***CodingFactory Testbed***



Ασκήσεις - Νέο Package

Προγραμματισμός με Java



- Δημιουργήστε νέο package ***gr.aueb.cf.ch1***



Ασκήσεις (1)

Προγραμματισμός με Java

Γράψτε ένα πρόγραμμα, νέα κλάση με όνομα **MenuApp** μέσα στο package `gr.aueb.gr.ch1`, που να εκτυπώνει το παρακάτω Μενού (χωρίς το κόκκινο πλαίσιο):

Επιλέξτε μία από τις παρακάτω επιλογές:

1. Εισαγωγή
2. Διαγραφή
3. Αναζήτηση
4. Ενημέρωση
5. Έξοδος

Δώστε αριθμό επιλογής:



Ασκήσεις (2)

- Γράψτε ένα πρόγραμμα, νέα κλάση με όνομα **InitialsApp** μέσα στο *package gr.aueb.cf.ch1*, που να εκτυπώνει τα αρχικά του ονόματός σας με κεφαλαία γράμματα, με αστεράκια

```
"C:\Program Files\Amazon Corretto\jdk11
*****          *
*          *          *  *
*          *          *  *
*****          *****
*          *          *      *
*          *          *      *
*****          *          *
```

Process finished with exit code 0



Ασκήσεις (3)

- Γράψτε ένα πρόγραμμα που δηλώνει δύο μεταβλητές ακεραίων *num1* και *num2* και αποθηκεύει το αποτέλεσμα της πρόσθεσης *num1 + num2* στην μεταβλητή *sum*. Στη συνέχεια εκτυπώνει με `println` το εξής κείμενο: Το αποτέλεσμα της πρόσθεσης είναι ίσο με [το αποτέλεσμα της πρόσθεσης]
- *Δοκιμαστικά Δεδομένα*
 - *num1 = 19 και num2 = 30*
 - *Αναμενόμενα Αποτέλεσμα: 49*



Τι να προσέξετε στις Ασκήσεις

Προγραμματισμός με Java

- Όπως αναφέρθηκε, δημιουργήστε ένα project με όνομα **CodingFactoryTestbed** ή **CodingFactory6** και ένα package με όνομα **gr.aueb.cf.ch1**. Μέσα στο ch1 θα κάνετε τις ασκήσεις της 1^{ης} εβδομάδας
- Να προσέχετε τις συμβάσεις. Ονόματα κλάσεων ξεκινάνε με κεφαλαία, ονόματα μεταβλητών ξεκινάνε με πεζά
- Να προσέχετε τη στοίχιση. Μετά από κάθε { πάμε στην επόμενη γραμμή με 1 tab δεξιά. Το IntelliJ αυτόματα τοποθετεί το σημείο εισαγωγής