

Curs pregatitor Java Fundamentals

by Andrei Toma & Cătălin Boja

Obiectiv: Dobandirea cunostintelor teoretice si practice de baza aferente programarii orientate obiect folosind limbajul de programare Java

Topic-uri pentru o perioada de invatare de 8 saptamani (cca 64 ore):

- Concepte de baza ale masinii virtuale Java - JVM si specificatii.
- JDK, JRE, compilare din linia de comanda si dezvoltare folosind un IDE (Netbeans).
- Conceptele programarii orientate obiect, clase, obiecte, derivare, interfete, polimorfism (late binding) in Java.
- Cloning si mecanismul try-catch in Java.
- Java Generics si Java Collection Framework.
- I/O Streams - byte/char - input/output streams.
- Lucru cu baze de date (JDBC) in aplicatii Java

Software necesar:

Pentru testarea exemplor din carti si tutoriale se instaleaza jdk 1.6 si preferabil netbeans.

Ghid de instalare jdk 1.6 (Java SE Development Kit (JDK) 6 Update 10)

- Sursa de download: <http://java.sun.com/javase/downloads/index.jsp>
- Ghid de instalare: <http://java.sun.com/javase/6/webnotes/install/jdk/install-windows.html>

Ghid de instalare Netbeans:

- Sursa de download (Netbeans 6.1):
<http://www.netbeans.org/downloads/start.html?platform=windows&lang=en&option=java>
- Ghid de instalare <http://www.netbeans.org/community/releases/61/install.html>

Recomandari:

Pentru realizarea exemplurilor si pentru o buna colaborare in cadrul echipei va recomandam sa va obisnuiti cu un mod standard de notare a variabilelor si a functiilor: naming convention(CamelCase, <http://en.wikipedia.org/wiki/CamelCase>, significant names), code indenting, package structure etc.

Surse de documentare (si care este utilitatea lor):

Sun Java Tutorial:

<http://java.sun.com/docs/books/tutorial/index.html>

Descriere. Contine descrierea principalelor notiuni legate de limbajul Java. Este tutorialul oficial Sun si este destul de scurt si foarte clar, motiv pentru care va functiona ca instrument principal de invatare.

Sun Certification Preparation

<http://java.sun.com/docs/books/tutorial/extra/certification/index.html>

Descriere. Contine documentatia necesara pregatirii pentru obtinerea certificatii Sun pentru JavaSE. Va fi consultat atunci cand tema respectiva nu este tratata suficient de clar in tutorial sau cand se doreste o tratare mai amanuntita.

Ca referinta suplimentara, **Core Java, volumul 1**, pe care o gasiti aici

<http://www.esnips.com/doc/267812fc-ce25-4b91-85ee-b4b7a1e07f8a/core-java-tm-volume-i-fundamentals-8th-edition.9780132354769.28647>

Cartea are dezavantajul dimensiunii, deci se citeste selectiv, doar pentru temele care sunt neclare.

O alta carte pe care o recomandam este: **Learning Java, 3rd Edition**, By Jonathan Knudsen, Patrick Niemeyer, O'Reilly pe care o gasiti aici

www.acs.ase.ro/plagiarism/Learning_Java_2005.zip

Prin prisma topic-urile, sunt recomandate capitolele 1 - 12. Cartea este destul de ok si ca mod de prezentare a elementelor teoretice si prin prisma exemplurilor.

Tutorial Java Catalin Boja

ce este disponibil pe <http://www.itcsolutions.eu/2010/12/22/tutorial-java-6-contents/> .

1. Introductory elements

- Exploring Java
- Java development with NetBeans

Detaliere sectiunea 1:

- mediul java
(<http://java.sun.com/docs/books/tutorial/essential/environment/index.html>) ; util pentru a intelege mai bine anatomia unei aplicatii java
- concepte ale platformei Java: masina virtuala, compilare din linia de comanda, SDK
- structura unei aplicatii Java - Hello World
- lucru in linia de comanda sau in IDE (Netbeans)
- din cartea **Learning Java, 3rd Edition** se citesc capitolele 1,2,3

2. Fundamental Java concepts (2 weeks)

- Object oriented programming basics
- Flow control
- Operators
- Data types
- Packages

Detaliere sectiunea 2:

- concepte fundamentale ale programarii orientate pe obiect, fundamente ale limbajului
- o aplicatie simpla:
<http://java.sun.com/docs/books/tutorial/getStarted/cupojava/index.html> (sa se creeze si in netbeans pentru a se intelege cum se construiesc/ruleaza proiectele) si explicatia ei
<http://java.sun.com/docs/books/tutorial/getStarted/application/index.html>
- concepte ale programarii orientate pe obiect (OOP)
<http://java.sun.com/docs/books/tutorial/java/concepts/index.html> . Incercati sa creati o aplicatie simpla de test pentru verificarea intelegerii conceptelor (clasa, obiect, mostenire, interfete, pachete, constructori de clase). Sfatul meu este sa porniti cu o aplicatie simpla si sa o extindeti
- elementele fundamentale ale limbajului java
<http://java.sun.com/docs/books/tutorial/java/nutsandbolts/index.html> . Incercati sa vedeti care este corespondenta dintre elementele fundamentale ale limbajului Java si cele ale limbajului pe care il cunoasteti sau (in cazul in care Java este primul limbaj de programare invatat) sa intelegeti care sunt conceptele care stau la baza stilului de programare imperativa (cum este Java), adica executarea unui set de instructiuni

consecutive, cu modificarea fluxului de executie. Elementele care trebuie cunoscute sunt cele tipice pentru un limbaj de programare: ce tipuri de date sunt folosite, ce sunt variabilele, ce operatori sunt folositi intre ele, cum se evalueaza expresiile, cum se grupeaza instructiunile in blocuri, controlul fluxului de executie (prin instructiuni, "if", "for" etc.)

- clase si obiecte <http://java.sun.com/docs/books/tutorial/java/javaOO/index.html> . Cum se construiesc clase si obiecte, functii si attribute, cum se returneaza o valoare printr-o functie, cum se acceseaza membrii unei clase (controlul accesului, cuvantul cheie this).Nested classes sunt optionale, dar recomandate. Tipul enumerare. Adnotarile sunt optionale, dar recomandate. Se citeste insa un tutorial de javadoc : <http://java.sun.com/j2se/javadoc/writingdoccomments/index.html>
- mostenire si interfete <http://java.sun.com/docs/books/tutorial/java/landl/index.html> . Cum se reutilizeaza codul prin mostenire, interfetele ca modalitate de evitare a mostenirii multiple, clase abstracte, suprascrierea metodelor si a campurilor, cuvantul cheie super.
- pachete <http://java.sun.com/docs/books/tutorial/java/package/index.html> . Ce sunt si cum se folosesc; organizarea surselor Java.
- din cartea **Learning Java, 3rd Edition** se citesc capitolele 4,5,6,7,10

3. Fundamental classes (2 weeks)

- Numbers and arrays
- Exceptions
- Streams and Input/ Output operations
- Regular expressions
- Concurrency basics
- Generics
- Collections

Detaliere sectiunea 3

- exceptii (<http://java.sun.com/docs/books/tutorial/essential/exceptions/index.html>) .
 - ce e o exceptie, try, catch, finally, throw (ganditi-va la ce folosesc exceptiile si de ce imbunatatesc calitatea codului;
 - de ce ar trebui sa existe exceptii in viziunea unui sistem bazat pe eveniment;
 - este preferabil sa se prinda exceptii generale sau specifice? i.e. Exception vs. NumberFormatException).
 - incercati o scurta aplicatie cu conversii de la siruri de caractere la numere pentru a vedea functionarea exceptiilor.

- operatii de intrare/iesire (<http://java.sun.com/docs/books/tutorial/essential/io/index.html>) .
 - input pe streamuri, diversele tipuri de streamuri, stream piping;
 - streamuri de fisiere secventiale si cu acces direct (random access)
- concurenta (<http://java.sun.com/docs/books/tutorial/essential/concurrency/index.html>).
 - clasa thread, interfata runnable;
 - sincronizare si probleme ale concurentei (erori de acces, locking etc.).
 - cat de flexibile sunt threadurile ca mecanism de concurenta?
 - de ce avem nevoie de concurenta?
 - de ce avem o clasa si o interfata pentru a construi threaduri?
- optional, expresii regulate (<http://java.sun.com/docs/books/tutorial/essential/regex/index.html>) ; utilitatea expresiilor regulate in Java este dependenta de cunoasterea expresiilor regulate ca atare, deci daca nu stiti expresii regulate nu e cazul sa cititi asta deocamdata
- tipuri generice <http://java.sun.com/docs/books/tutorial/java/generics/index.html> . Ce sunt tipurile generice, cum se declara, cum se folosesc, metode generice, constructori generici, limitarea tipurilor acceptate pentru un tip generic.
- tema (optional):
 - o aplicatie simpla cu streamuri si exception handling (cititi un fisier bmp si afisati campurile din header);
 - o aplicatie simpla cu threaduri
- din cartea **Learning Java, 3rd Edition** se citesc capitolele 8,9,12

4. Collections (2 weeks)

- Interfaces
- Implementations
- Algorithms

Detaliere sectiunea 4

- interfete (<http://java.sun.com/docs/books/tutorial/collections/interfaces/index.html>)
 - la ce sunt utile diferitele feluri de colectii?
 - care este diferenta dintre Collection si Map?; la ce ar fi util Map si la ce ar fi util Collection?
 - Map, Collection, Set, List, Queue
 - foreach si iteratori
 - operatii bulk, conversii la Array
 - operatii specifice principalelor tipuri de colectii

- implementari
(<http://java.sun.com/docs/books/tutorial/collections/implementations/index.html>)
- algoritmi (<http://java.sun.com/docs/books/tutorial/collections/algorithms/index.html>)
- din cartea **Learning Java, 3rd Edition** se citesc capitolele 11

5. Other (2 weeks)

- Events and event listeners - Event Listeners
(<http://java.sun.com/docs/books/tutorial/uiswing/events/index.html>)
- Background tasks - Task-uri executate in background
(<http://java.sun.com/docs/books/tutorial/uiswing/concurrency/index.html>)
- JDBC <http://java.sun.com/docs/books/tutorial/jdbc/index.html>
- javadoc, adnotations (for comments)
<http://java.sun.com/j2se/javadoc/writingdoccomments/index.html>
- din cartea **Learning Java, 3rd Edition** se citesc capitolele 16,17,18,19