



Școala
informală
de IT

Java Basics – Curriculum

Școala Informală de IT

Educație Informală S.A.

Tel: +40744-679.530

Web: www.scoalainformala.ro / www.informalschool.com

E-mail: info@scoalainformala.ro



Content

1. Software development - short intro; Programming basics
2. Intro to Java
3. Intro to GIT
4. OOP concepts
5. Java OOPs Concepts – I
6. Using Eclipse for development
7. Unit Testing
8. Java OOPs Concepts - II
9. Object containers
10. Exceptions
11. Generics
12. IO
13. Enumerations; Annotations
14. Concurrency
15. Databases & JDBC
16. Java WEB
17. Team Project
18. Interview Preparation



1. Software development - short intro; Programming basics

- Software development importance
- Software development process
- Basic principles and terminology
- Data structures
- Operators
- Expressions, Statements, and Blocks
- Control Flow Statements
- Functions – how to solve a real life problem by implementing an application

2. Intro to Java

- History
- Key concepts
- Java ecosystem
- Java usage - types of applications
- Application structure and elements
- Syntax
- Tools chain: jdk, jre, java, javac, jar, etc
- Build and run a simple program
- Naming conventions
- Java doc
- Java api documentation



3. Intro to GIT

- What is GIT?
- How to use it
- Homework structure on git
- Homework process

4. OOP concepts

- Data abstraction & communication through messages
- Encapsulation
- Inheritance
- Polymorphism
- OOP advantages
- UML basics

5. Java OOPs Concepts - I

- Everything is an object
 - Creating objects
 - Using objects
 - Fields, methods, arguments, return values
- Initialization and cleanup
 - This keyword
 - Default constructor
 - Constructor initialization
 - Member initialization



- Cleanup: finalization and GC
- Primitive types
- (Un)Boxing

6. Using Eclipse for development

- Intro to Eclipse
- Write, compile and execute a program in Eclipse
- Coding style
- Debug

7. Unit Testing

- Unit testing concepts
- Implementing unit tests with JUnit
- TDD intro
- Best practices

8. Java OOPs Concepts - II

- Access control
 - Reusing classes
 - Inheritance
 - Composition



- Super and final keywords
- Interfaces
- Abstract classes
- Runtime polymorphism
- Upcasting and down-casting
- Static and dynamic binding
- Inner, nested and anonymous classes

9. Object containers

- Basic concepts
- Arrays
- List, Set, Map, Iterator, Stack, Queue
- Equals and hashCode
- Collections framework
- Best practices

10. Exceptions

- The need and the usage
- Syntax
- Checked vs unchecked exceptions
- Best practices

11. Generics

- Comparison with C++
- Simple generics



- Generic interfaces
- Generic methods
- Type erasure
- Wildcards
- Best practices

12. IO

- InputStream & OutputStream
- Readers & Writers
- IO decorators
- Standard IO
- NIO
- Charsets & encoding
- Best practices

13. Enumerations; Annotations

- Enumerations:
 - Enum keyword
 - Using enums in switches
 - Adding methods
 - EnumSet and EnumMap
- Annotations:
 - Basic syntax
 - Writing annotation processors



14. Concurrency

- Basic threading
 - Thread
 - Runnable
 - Executor
 - Priority
 - Daemon threads
- Share resources
- Cooperation between tasks
- New concurrency components:
 - Latches
 - Future tasks
 - Barriers
 - Blocking queues
 - Semaphore
- Best practices

15. Databases & JDBC

- Intro to Databases
- Intro to SQL
- JDBC Concepts
- Connection
- Statement
- PreparedStatement
- ResultSet



- Datasource and connection pooling
- Transactions
- Best practices

16. Java WEB

- basic concepts: HTTP, HTTPS, HTML, CSS, JS
- servlets
- JSP

17. Team Project

- Intro to Spring
- Spring MVC
- Spring Boot
- Intro to Maven

18. Interview Preparation

- How to present oneself during an IT interview
- CV formatting tips