

## Fundamentals of Java

**Length:** 40 - 56 hours, depending on the practical part

**Required facilities:** VGA projector, white board, one workstation (per two programmers), Java JDK, Eclipse IDE

**Who can participate:**

- programmers who come from procedural programming languages world, for example C
- programmers who already use an object oriented programming language like C++ or C# and want to learn Java

**Course objectives:** learn the fundamentals of Java programming language in order to write stand alone programs (Java Standard Edition – J2SE), use of Eclipse as integrated development environment (IDE). The course prepares the attendees for Sun Certified Java Programmer (SCPJ), includes other basic subject as well

**Related courses:** Fundamentals of UML, Java Enterprise, Design Patterns

**Attendees' evaluation:** optional, during the training and/or a final test

**Course written support:** yes

**Description:** the course is focused on the Java language and on the main class packages. The major part of the time budget is dedicated to exercise Java, to learn the particularities and stereotypes used in Java, the practical aspects of object oriented programming.

**Contents:**

1. Introduction in Java programming: classes and objects, instance members and static members, inheritance and aggregation
2. Fundamentals of the language: lexical elements, declaring variables, references
3. Operators and assignments: operator precedence and associativity, type conversions, numerical promotion, operators, passing the parameters

4. Declarations and access control: arrays, methods, method overloading, constructors, packages, modifications, existence scope and members visibility, jar archives
5. Program control and exception handling, assertions
6. Object oriented programming: inheritance, OOP concepts, polymorphism, upcasting and downcasting, methods overriding, variables hiding, interfaces, constructors linking
7. Inner classes and interfaces: inner static classes and interfaces, non-static inner classes, locale classes, anonym classes
8. Object life cycle: garbage collection, object finalization, initializes, object initial state
9. Threads: java.lang.Runnable, java.lang.Thread, synchronization, thread states, priorities, starting/stopping, sleep/wake up, wait/notify, join, blocking types, termination, deadlock
10. Fundamental classes: Object, wrapper, String, StringBuffer
11. Collections and dictionaries: interfaces, implementations, collections, sets, collection personalization, particular issues
12. Files and streams: File class, byte and character streams, filters, objects serialization
13. AWT components: components and containers, GUI controls, menus
14. Layout Management: layout types, hierarchies of components
15. Event handling: hierarchies of events, event delegation model, event adaptors
16. Drawing: colors, fonts, drawing, clipping, AWT thread, events handling
17. Introduction in Swing: packages, components, look & feel, events handling
18. Applets: life cycle, interaction with the HTML context page text, security
19. Java 5.0, 6.0 – last issues added to Java: StringBuilder, generics, enumeration, autoboxing/unboxing, varargs, for/in, static imports, formatting