AMBIENT·IT

+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated 06/06/2025

Sign up

Java 23 Initiation : The Fundamentals

5 days (35 hours)

Presentation

Our Java programming fundamentals course offers a complete introduction to one of the world's most widespread programming languages. Specially designed for beginners, it will enable you to gradually acquire all the essential Java skills.

Java, renowned for its robustness, exceptional portability and vast ecosystem, regularly ranks among the most popular according to the TIOBE indexlanguages . Mastering Java is an essential asset for advancement in the IT sector.

During this course, you'll learn about object-oriented syntax, the IntelliJ IDEA development environment, object collection management and query execution. At the end of the course, you'll be able to design, build and deploy your own Java application.

You'll also explore advanced concepts such as exception handling, debugging tools and unit testing. You'll also explore the main developments in Java over the last ten years, so that you can adopt best practices regarding libraries and APIs.

The course is regularly updated to keep you abreast of the latest developments in the Java language (currently JDK 24).

Objectives

- Know the basic concepts of the JAVA language and master its syntax
- Using libraries and APIs
- Understand the concepts of object-oriented programming in Java
- Creating a Java application
- Manage code errors and use debugging tools
- Get to grips with what's new in Java

Target audience

- Developers
- Technical architects
- Computer application developers
- Project managers

Prerequisites

Basic knowledge of object programming. 10 days training if no knowledge.

Materials required

Intellij installed on his machine.

Java 24 Initiation Training Program

Introduction to Java Discover IntelliJ

- Overview of the JAVA language
- Discovering and getting to grips with IntelliJ IDEA
- Optimum initial configuration
- Discover Gradle
- Tips for optimizing your work environment
- Project architecture in Java
- StackTrace
- Using the debugger
- Practical workshop: Fully configure a Java project with IntelliJ and Gradle, then correct a deliberately faulty program using the integrated debugger.

Fundamental Java syntax

- Structure of a Java "Hello World" program
- Variables and primitive types
- Simple and multidimensional tables
- Operators and conditional expressions
- Loops, flow management
- Algorithms
- Input/output handling
- Practical workshop: Development of a simple Java console application for entering, calculating and displaying averages.

Object-oriented programming

- Basic syntax
  - Attribute
  - Pointer
  - Instance method
- Algorithms with objects
- Fundamental concepts
  - Polymorphism
  - Encapsulation
  - Manufacturers
  - Accessors
- Abstract methods and abstract classes
- Advanced concepts: method overloading and redefinition
- Practical workshop: "Warrior Game" Create a hierarchy of warriors (Archer, Brutos, Mage...) in Java, applying inheritance and polymorphism. Each warrior must have a specific attack.

Object collection and advanced data structures

- Picture and object collections
- ArrayList, LinkedList
- HashMap, HashSet
- Sorting algorithms and methods
- Advanced flow handling
- Generic types in Java
- Case study: "Advanced contact manager": Development of a Java application for dynamically adding, searching, sorting and deleting contacts via collections.

Exception handling and code robustness

- Exceptions in Java
- Managing errors effectively
- NullPointerException
- Advanced debugging tools
- Predefined exceptions
- Customized exceptions
- Practical workshop: Design an application capable of reading, writing and managing text files, while efficiently handling possible exceptions and errors.

External and network APIs

- External libraries with Gradle
- HTTP requests in Java
- JSON serialization and deserialization
- External REST APIs
- Simple asynchronous calls
- Practical workshop: retrieve and display weather forecasts from an external API using network requests and JSON parsing

Create your application

- Reading a class diagram
- Create a console game using several objects
- Relationship with a graphical interface
- Generating a graphical interface
- Introduction to inheritance with graphical components
- Introduction to interfaces using events on graphical components
- Modify data following an event and update the graphical interface
- Explaining and implementing SVM

Advanced concept

- Unit testing
  - Usefulness
  - Principles
- JUnit 5
- Methods, classes, exceptions
- Simple integration testing tools
- Quick introduction to Continuous Integration (CI/CD) for Java

Main new features in Java

- Main Java developments over the last 10 years
- Var, records, switch pattern matching
- Java modules
- Comparison between recent versions
- Practical workshop: Upgrading an old Java project to a recent version

Java developments and history

- Java 19 (September 2022)
  - Virtual Threads (Overview): lightweight, cost-effective threads
  - Pattern Matching for switch (Overview): better management of conditional structures
  - Record Patterns (Overview): simplifying data processing with records
- Java 20 (March 2023)
  - UTF-8 by default: for better standardization
  - Additional support for RISC-V architecture
  - Enhanced Reference Processing API
- Java 21 (September 2023, LTS)
  - Virtual Threads: lightweight threads for enhanced scalability
  - Structured Concurrency: structured competition management
  - Pattern Matching for switch: simplifying conditional instructions
  - Sequenced Collections : Standardized API for ordered collections
- Java 22 (March 2024)
  - Foreign Function & Memory API (final version): efficient interaction with native code
  - Unnamed Variables: anonymous variables to simplify expressions
  - Statements before super(): greater flexibility for constructors
  - Z Garbage Collector: ZGC collector upgrade

- Java 23 (September 2024)
  - Scoped Values: secure transmission of values between threads
  - String Templates: improving formatted strings
  - Structured Task Scope: simplifying concurrent tasks
  - Vector API: faster vectorized operations
- Java 24 (March 2025)
  - Primitive Types in Patterns: improving pattern matching with primitive types
  - Flexible Constructor Bodies
  - Quantum-Resistant Crypto: algorithms resistant to quantum attacks
  - Generational Shenandoah GC: new generation for Shenandoah Garbage Collector

Companies concerned

This course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the training to come, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical course: 60% Practical, 40% Theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, with brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire verifies the correct acquisition of skills.

Sanction

A certificate will be issued to each trainee who completes the course.