Introduction to Object/Java Programming

Hands-on course of 3 days - 21h
Ref.: IPJ - Price 2024: CHF1 870 (excl. taxes)
The price for the 2025 session dates may be revised

THE PROGRAMME

last updated: 01/2018

1) General Presentation

- Introduction.
- Fundamental principles of Object-Oriented Programming: Abstraction/Encapsulation.
- Fundamental principles of Object-Oriented Programming: Inheritance.
- General presentation: the Language.
- The Tools and the Library.
- Releases of Java.

2) Syntax, Types and Expressions

- Syntactic structure of a Java application.
- Syntax example of a simplified application.
- External view of a class: Use Syntax.
- Internal view of a class: Implementation Syntax.
- Concept of Type.
- Compared use of Primitive Types and Object Types.
- Simple use of Primitive Types: Integers, Floats, Characters and Booleans.
- Concept of Expression.
- Examples of Declarations: Variables and Constants.
- Compared declarations of Primitive Types and Object Types.
- Using Operators with Objects.
- Special case of Static Fields or Class Variables.
- Conversions between Primitive Types and Object Types.
- Code Conventions.

3) Methods and Statements

- Syntax of Method Calls.
- Class Methods and Instance Methods.
- Definition and Use of Methods.
- Method Overriding.
- Concept of Sub-Block.
- Categories of Statements.
- Main Control Statements: ?if?, ?while?, ?for?, ?return?, ?break?.

4) Using Abstraction

- Simple example of Use of an Object: Declaration, Instantiation or Creation, Delegation.
- Using Object Constructors.
- Using the Programming Interface of Objects: example of the Date class.
- A very useful class: the String class.
- Characteristics of Character Strings.
- Using the StringBuffer class: example of use of Method Overloading.

TRAINER QUALIFICATIONS

The experts leading the training are specialists in the covered subjects. They have been approved by our instructional teams for both their professional knowledge and their teaching ability, for each course they teach. They have at least five to ten years of experience in their field and hold (or have held) decision-making positions in companies.

ASSESSMENT TERMS

The trainer evaluates each participant's academic progress throughout the training using multiple choice, scenarios, hands-on work and more.
Participants also complete a placement test before and after the course to measure the skills they've developed.

TEACHING AIDS AND TECHNICAL RESOURCES

• The main teaching aids and instructional methods used in the training are audiovisual aids, documentation and course material, hands-on application exercises and corrected exercises for practical training courses, case studies and coverage of real cases for training seminars.
• At the end of each course or seminar, ORSYS provides participants with a course evaluation questionnaire that is analysed by our instructional teams.
• A check-in sheet for each half-day of attendance is provided at the end of the training, along with a course completion certificate if the trainee attended the entire session.

TERMS AND DEADLINES

Registration must be completed 24 hours before the start of the training.

ACCESSIBILITY FOR PEOPLE WITH DISABILITIES

Do you need special accessibility accommodations? Contact Mrs. Fosse, Disability Manager, at psh-accueil@ORSYS.fr to review your request and its feasibility.

5) Using Inheritance

- Reminder of the Inheritance Principle and Terminology.
- Using Inheritance.
- Example of Inheritance Graph.
- The Object class and Genericity.
- Using Polymorphism.
- Specialization of a Polymorphic Reference.
- Typing of References / Typing of Objects.
- Behavior of Methods and Typing.
- Genericity of Collection Classes: example of the Vector class.
- New Features in Java 5 (Tiger): Generics.

6) Using Interface

- Implicit/Explicit Interface of a Class.
- Syntax of Explicit Interfaces.
- Use case of Interface References: Flexibility, Scope Reduction, Polymorphism.
- Example of Implementation of Multiple Interfaces.
- Synthesis of the benefit of Interfaces for Methods.
- Using Interfaces for Constants.
- Advanced examples of Use of Interfaces.

7) Developing Classes

- Methodological Approach; Static, Dynamic and Business Analysis.
- UML Notation: Class, State and Sequence Diagrams.
- Class Skeleton: Basic Components, Tools of Code Automatic Generation.
- Additional information about Access Rights.
- Organizing Packages.
- Constraints related to Packages.
- Implementing Constructors.
- Default Constructor.
- Additional information about the Implementation of Constructors.
- Self Reference ?this?.
- Static Fields and Methods.
- ?main? Method

8) Developing Interfaces

- Syntax of Interfaces, case of Constants.
- Definition of Interfaces for Methods.
- Implementing and Extending Multiple Interfaces.
- Partial Implementation of Interface.
- Examples and additional information about the Use of Interfaces.

9) Developing Derived Classes

- Reminder of the Principles.
- Methodological Approach for a Decomposition in Classes.
- Abstract Methods and Classes.
- Interfaces and Abstract Classes.
- Inheritance and Access Rights of Fields.
- Inheritance and Call of Constructors.
- Overriding and Overloading.

10) Exceptions

- Principles and Event Sequence.
- Catching and Handling an Exception.
- Throwing an Exception.
- Unchecked Exceptions.

- Simple example with Exception Handling.

DATES

REMOTE CLASS
2024 : 16 Dec

2025 : 12 Feb, 12 May, 18 Aug,
10 Dec