Training: Capstone Courseware
103 Java Programming

TRAINING GOALS:

Version 8.0

This course teaches programming in the Java language -- i.e. the **Java Standard Edition** platform. It is intended for programmers with experience in languages other than Java, but who may or may not have any previous Java experience. It focuses on procedural and structured coding skills first, and then offers meticulous, in-depth coverage of object-oriented concepts and how to apply them to Java software design and development. The latter part of the course moves from these basic skills into key parts of the **Java SE Core API**, including collections and logging, and introduces features of functional programming, new to the language as of **Java 8**, including functional interfaces, lambda expressions, and streams.

This revision of the course targets the Java 8 language and Core API.

Students come to Java from a wide range of backgrounds, and this course is designed to be as flexible as possible over the upper end of that range. Specifically:

- Experienced C and C++ programmers will find this course a very good fit and if anything will find that they complete it in a little less than the full five-day timeline.
- Those with experience in languages less like Java, such as Visual Basic, ASP and other Web-scripting languages, and other pseudo-object-oriented languages may need more time in the early going, and this course covers its introductory topics in good depth and offers many optional and "challenge" labs to support this.
- Less experienced programmers or those coming from non-structured languages -- such as COBOL, PL/1, or 4GL tools -- will probably not cover the whole course in a week, and may want to pursue an abbreviated version at a slower pace. This too is quite feasible, but this audience may also want to consider course 102 **Introduction to Java Programming**, for a more relaxed pace through the early material.

Learning Objectives

- Chiefly, learn to program effectively in the Java language.
- Understand the Java software architecture, and the design decisions which make Java software portable, efficient, secure and robust.
- Learn how to configure a simple Java development environment.
- Know the grammar, data types and flow control constructs of the Java language for simple procedural programming.

Compendium Education Center Ltd.
ul. Tatarska 5, 30-103 Kraków, tel.: (12) 298 47 77
ul. Bielska 17, 02-394 Warszawa, tel.: (22) 417 41 70

BNP Paribas Bank Polska S.A.
EUR: PL 79 1600 1462 1853 6451 3000 0006
USD: PL 52 1600 1462 1853 6451 3000 0007

- Understand Java as a purely object-oriented language, and implement software as systems of classes.
- Implement and use inheritance and polymorphism, including interfaces and abstract classes.
- Design appropriate exception handling into Java methods, and use the logging API appropriately.
- Use Java as a functional language, making appropriate choices of tools including inner classes, functional interfaces, method references, and lambda expressions.
- Use the Stream API for efficient processing of data sets.

CONSPECT:

- The Java Environment
  - Overview of Architecture
  - Forms for Java Software
  - J2SE, J2EE, and J2ME Platforms
  - Java Virtual Machine
  - The Core API
  - Java Runtime Environment
  - Java Developer's Kit
  - Java Class Path
  - Classes
  - Built-In Streams and Command-Line Parameters
- Language Fundamental
  - Source File Format
  - Application Classes
  - Code Grammar and Expressions
  - Identifiers
  - Literals
  - Operators
  - Calling Methods
  - Variable Parameter Lists ("varargs")
- Data Types
  - Strict Type Checking
  - Primitive Types
  - Numeric Types
  - Characters and Booleans
  - Enumerations

Compendium Education Center Ltd.
ul. Tatarska 5, 30-103 Kraków, tel.: (12) 298 47 77
ul. Bielska 17, 02-394 Warszawa, tel.: (22) 417 41 70
BNP Paribas Bank Polska S.A.
EUR: PL 79 1600 1462 1853 6451 3000 0006
USD: PL 52 1600 1462 1853 6451 3000 0007

    - Type Conversion
    - Formatted Output
    - Object References
    - Comparing and Assigning References
    - Strings
    - Arrays
- Flow Control
    - The main Method
    - Calling and Returning from Methods
    - Conditional Constructs
    - Looping Constructs
    - Processing Arrays
    - Looping and Enumerations
    - Processing Varargs
    - The Flow-Control Operator
    - Break and Continue
    - Recursion
- Object-Oriented Software
    - Complex Systems
    - Abstraction
    - Classes and Objects
    - Responsibilities and Collaborators
    - UML
    - Relationships
    - Visibility
- Classes and Objects
    - Java Classes
    - Constructors and Garbage Collection
    - Naming Conventions and JavaBeans
    - Relationships Between Classes
    - Using this
    - Visibility
    - Packages and Imports
    - Overloading Methods and Constructors
    - JARs
- Inheritance and Polymorphism in Java

Compendium Education Center Ltd.
ul. Tatarska 5, 30-103 Kraków, tel.: (12) 298 47 77
ul. Bielska 17, 02-394 Warszawa, tel.: (22) 417 41 70
BNP Paribas Bank Polska S.A.
EUR: PL 79 1600 1462 1853 6451 3000 0006
USD: PL 52 1600 1462 1853 6451 3000 0007

- UML Specialization
- Extending Classes
- Using Derived Classes
- Type Identification
- Compile-Time and Run-Time Type
- Polymorphism
- Overriding Methods
- The @Override Annotation
- Superclass Reference

- Using Classes Effectively
  - Class Loading
  - Static Members
  - Statics and Non-Statics
  - Static Initializers
  - Static Imports
  - Prohibiting Inheritance
  - Costs of Object Creation
  - Strings and StringBuffers
  - Controlling Object Creation
  - Understanding Enumerated Types
  - Stateful and Behavioral Enumerations
- Interfaces and Abstract Classes
  - Separating Interface and Implementation
  - UML Interfaces and Realization
  - Defining Interfaces
  - Implementing and Extending Interfaces
  - Abstract Classes
- Collections
  - Dynamic Collections vs. Arrays
  - UML Parameterized Type
  - Generics
  - Using Generics
  - The Collections API
  - The Collectionand ListInterfaces
  - The ArrayListand LinkedListClasses
  - Looping Over Collections: Iterable

Compendium Education Center Ltd.
ul. Tatarska 5, 30-103 Kraków, tel.: (12) 298 47 77
ul. Bielska 17, 02-394 Warszawa, tel.: (22) 417 41 70
BNP Paribas Bank Polska S.A.
EUR: PL 79 1600 1462 1853 6451 3000 0006
USD: PL 52 1600 1462 1853 6451 3000 0007

- Collecting Primitive Values: Auto-Boxing
- Using Wildcards with Generic Types
- Iterators and the IteratorInterface
- Maps and the MapInterface
- Sorted Collections
- The SortedSetand SortedMapInterfaces
- The Collections Class Utility
- Algorithms
- Conversion Utilities

- Exception Handling and Logging
  - Reporting and Trapping Errors
  - Exception Handling
  - Throwing Exceptions
  - Declaring Exceptions per Method
  - Catching Exceptions
  - The finally Block
  - Catch-and-Release
  - Chaining Exceptions
  - try-with-resources
  - Logging
  - The Java SE Logging API
  - Loggers
  - Logging Levels
  - Handlers
  - Configuration
  - Best Practices
- Nested Classes
  - Nested Classes
  - Static Classes
  - Inner Classes
  - Relationship with the Outer Object
  - Local Classes
  - Enclosing Scope
  - Anonymous Classes
- Functional Programming
  - Passing Behavior as a Parameter

Compendium Education Center Ltd.
ul. Tatarska 5, 30-103 Kraków, tel.: (12) 298 47 77
ul. Bielska 17, 02-394 Warszawa, tel.: (22) 417 41 70

BNP Paribas Bank Polska S.A.
EUR: PL 79 1600 1462 1853 6451 3000 0006
USD: PL 52 1600 1462 1853 6451 3000 0007

- Inner Classes
- Functional Interfaces
- Built-In Functional Interfaces
- Lambda Expressions
- Scope and Visibility
- Deferred Execution
- Method References
- Creational Methods
- Designing for Functional Programming
- Default Methods

- Streams
  - The Stream Processing Model
  - Streams
  - Relationship to Collections
  - Advantages and Disadvantages
  - Iterating, Filtering, and Mapping
  - Primitive-Type Streams
  - Aggregate Functions and Statistics
  - Sorting
  - Generating, Limiting, and Reducing
  - Finding and Matching
  - Grouping
  - Flattening and Traversing
  - Sequential vs. Parallel Processing

REQUIREMENTS:

No prior Java experience is required, but students must be experienced programmers in another third-generation (high-level) language. See the overview for suggestions about pace and scope for different backgrounds.

Difficulty level

CERTIFICATE:

The participants will obtain certificates signed by Capstone Courseware.

Compendium Education Center Ltd.
ul. Tatarska 5, 30-103 Kraków, tel.: (12) 298 47 77
ul. Bielska 17, 02-394 Warszawa, tel.: (22) 417 41 70
BNP Paribas Bank Polska S.A.
EUR: PL 79 1600 1462 1853 6451 3000 0006
USD: PL 52 1600 1462 1853 6451 3000 0007

TRAINER:

Authorized Capstone Courseware Trainer.

Compendium Education Center Ltd.
ul. Tatarska 5, 30-103 Kraków, tel.: (12) 298 47 77
ul. Bielska 17, 02-394 Warszawa, tel.: (22) 417 41 70

BNP Paribas Bank Polska S.A.
EUR: PL 79 1600 1462 1853 6451 3000 0006
USD: PL 52 1600 1462 1853 6451 3000 0007