

JAV500 : Advanced Java Programmeren

Code : JAV500

Duur : 4 dagen

Categorie : Java

Doelgroep :

Ervaren Java developers die meer geavanceerde aspecten van Java willen leren.

Voorkennis :

Kennis van de syntax van de Java taal en fundamentele ervaring met programmeren in Java is vereist.

Uitvoering :

De theorie wordt behandeld aan de hand van presentatie slides en wordt afgewisseld met oefeningen. Demo's dienen ter verheldering van de theorie.



Advanced Java Programming



Inhoud :

In deze cursus Advanced Java Programmeren komen een reeks geavanceerde aspecten van Java aan de orde. Zo wordt ingegaan op de verschillende manieren waarop Java applicaties via het netwerk kunnen communiceren. In het bijzonder wordt hierbij ingegaan op sockets en Remote Method Invocations (RMI). Verder komen de geavanceerde multithreading en synchronization mogelijkheden aan bod zoals die zijn geïntroduceerd in het concurrency package sinds Java 5. Ook wordt aandacht besteed aan de JMX API voor het local en remote managen en monitoren van Java applicaties. Verder staat reflection, waarmee gecompileerde Java classes softwarematig kunnen worden geanalyseerd, op met programma en komen diverse aspecten het verbeteren van de Java performance aan bod. Vervolgens wordt database access met Java Database Connectivity (JDBC) behandeld en komt aan de orde hoe Java applicaties via het Java Native Interface (JNI) in C en C++ geschreven code kunnen aanroepen. Ook de new IO API passeert de revue en na een behandeling van diverse aspecten van memory management en logging staat tenslotte de interactie tussen Java en XML op het programma. Hierbij wordt zowel aandacht besteed aan de streaming API (Stax) waarbij de applicatie de controle heeft over de XML benadering als aan de statische Java API for XML Binding (JAXB) waarbij vooraf Java classes voor specifieke XML worden gegenereerd. Ook Web Services die sinds Java 6 mogelijk zijn in de Standard Edition worden besproken. Deze cursus is een goede voorbereiding op het Java 7 Programmer Exam Part II (1Z0-804). In combinatie met de cursus Java Programming Fundamentals is de cursus Advanced Java Programmeren ook een goede voorbereiding op het Java 7 of Java 8 Programmer Exam Part I (1Z0-803) of (1Z1-808).

Module 1 : Java Networking

- Java Networking Intro
- Network Protocols
- Using URL Class
- Reading from an URL
- InetAddress Class
- Sockets
- Socket Communication
- Socket Types
- Socket Ports
- ServerSocket Class
- Socket Class
- Datagram Sockets
- DatagramPacket Class
- Multicast Sockets
- Socket Programming

Module 2 : Concurrency

- Concurrency Package
- Concurrency Package Benefits
- Contents Concurrency Package
- Task Scheduling Framework
- Executor Interface
- ExecutorService
- Executors Class
- Callable and Futures
- ScheduledExecutorService
- Synchronizers
- Semaphores
- CountdownLatch
- CyclicBarrier
- Exchanger
- Concurrent Collections
- BlockingQueue Interface
- Lock Interface
- Atomic Variable

Module 3 : Java Management Extensions

- What is JMX?
- JMX Goal
- JMX Usage
- Managed Beans
- MBean Flavors
- JMX Architecture Levels
- JMX Architecture
- MBeans Instrumenting JVM
- Accessing JMX Agent
- Standard MBeans
- Naming MBeans
- MBean Server
- MBean Server Interface
- Registering MBeans
- JMX Notifications
- Notification Class
- Notification Listeners

Module 4 : Reflection

- What is Reflection?
- Reflection Classes
- Class Loading
- Creating Objects
- Reflection Methods in Class
- Field Class
- Field Class Usage
- Constructor Class
- Constructor Class Usage
- Method Class
- Method Class Usage
- AccessibleObject Class
- Dynamic Proxies
- Invocation Handler

Module 5 : Performance

- Influences on Performance
- History of Java Performance
- JIT Compilation
- Hotspot JVM
- Garbage Collection
- Benchmarks
- Monitoring, Profiling, Tuning
- String Types
- Buffered and New I/O
- Synchronization
- Concurrency Package
- Primitives versus Wrappers
- Collections
- Exception Handling
- Serialization
- Native methods
- Lazy Loading
- Object Reuse

Module 6 : Remote Method Invocation

- What is RMI?
- RMI Characteristics
- RMI Architecture
- Remote Interface
- Stubs and Skeletons
- Remote Object
- Remote Object Registration
- RMI Client
- Exporting Objects
- Locating Remote Objects
- RMI Control Flow
- Serialization
- Parameter Passing
- Dynamic Class Loading
- Activation

Module 7 : Database Access with JDBC

- JDBC Architecture
- JDBC Drivers and URL's
- Database Connections
- Executing Statements
- Querying Databases
- Update Statements
- Retrieving Results
- Handling Errors
- Prepared Statements
- Database Metadata
- Transactions
- Commit and Rollback
- Rowset Interfaces
- Using RowsetProvider

Module 8 : Memory Management

- JVM Internal Architecture
- Java Memory Management
- Object Lifecycle
- Strong Object References
- Invisible and Unreachable
- Circular References
- Garbage Collection
- Generational GC
- Heap Space Organization
- GC Algorithms
- Serial Collector
- Parallel Collector
- Reference Objects
- Soft and Weak References
- Phantom References
- Finalization

Module 9 : Java Native Interface (JNI)

- Java Native Interface
- Java to C
- C to Java
- JNI Boundary
- Header Files
- Required Parameters
- Java Language
- Writing Native Methods
- Declaring Native Methods
- JNIEnv Interface Pointer
- Mapping Table
- Accessing Java Strings
- JNI Functions
- Accessing Arrays

Module 10 : Java Logging

Logging
Logging in Java
Java Util Logging
Levels and Log Methods
Logging Configuration
log4j Characteristics
log4j Configuration
Loggers
Logger Output Hierarchy
Inheriting Logging Levels
Logger Names
Log Levels
Appenders and Layouts
Log Analyzer Tools

Module 13 : StaX

Streaming API for XML
Building a Dom Tree
SAX Callbacks
Pull versus Push Parsing
Advantages of Pull Parsing
Why StaX?
Iterator API
XMLEventReader
XMLEventWriter
Cursor API
XMLStreamReader
XMLStreamWriter
Choosing considerations

Module 11 : New IO

What is NIO?
Synchronous I/O Processing
Asynchronous I/O Processing
Working with Buffers
IO Channels
Selectable Channels
Selectors
Selection Keys
Character Sets
Using Path Class
Directory Traversing
PathMatcher class
Using WatchService

Module 14 : Web Services

What is a Web Service?
Web Service Stack
What is SOAP?
What is WSDL?
What is JAX-WS
Web Service Support on Java 7
Service Programming Model
Publishing an Endpoint
Client Programming Model

Module 12 : Java API XML Binding (JAXB)

XML Processing Options
What is JAXB?
JAXB versus DOM and SAX
JAXB Design Goals
Building JAXB Applications
JAXB Architecture
JAXB Binding Life Cycle
Role of Binding Compiler
XML to Java Mapping Rules
Mapping of XML Schema Types
Binding Elements and Attributes
Named Complex Types
Customized Mappings
Adapter Classes
JAXB Annotations Types
JAXB API