

Java ist eine plattformunabhängige Programmiersprache, die ursprünglich von der Firma Sun Microsystems, entwickelt wurde. Java wird in allen IT-Bereichen eingesetzt und ist für unzählige Betriebssysteme und Plattformen, vom mobilen Telefon bis hin zur Echtzeit-Großrechneranlage geeignet.

### Ihr Nutzen

In diesem Seminar werden die Syntax der Programmiersprache Java, die objektorientierte Programmierung mit Java, Ausnahmen, Datei-Ein-/Ausgabe (E/A) und Threads behandelt.

### Preis pro Teilnehmer

EUR 2850,- exklusive der gesetzlichen MwSt.

### Seminardauer

5 Tag(e)/Day(s)

### Seminarinhalte

#### 1. Tag

- \* Java-Syntax und -Klassen - Wiederholung
- Einfache Java-Klassen
- Java-Felder, -Konstruktoren und -Methoden
- Modellieren von Objekten mit Java-Klassen
- Package- und Importanweisungen
  
- \* Kapselung und Polymorphie
- Kapselung im Design von Java-Klassen
- Unveränderbarkeit
- Erstellen von Unterklassen, Überladen von Methoden
- Methoden mit variablen Argumenten

#### \* Design von Java-Klassen

- Zugriffsmodifizierer private, protected und public
- Überschreiben von Methoden
- Überladen von Konstruktoren
- Zugriffsmodifizierer private, protected und public
- Virtueller Methodenaufruf, Polymorphie
- Casting von Objektreferenzen
- Überschreiben von Objektmethoden

#### 2. Tag

- \* Erweitertes Design von Klassen
- Abstrakte Klassen und Typpeneralisierung
- Modifizierer static und final
- Best Practices für Feldmodifizierer
- Entwurfsmuster Singleton
- Design abstrakter Klassen
- Verschachtelte Klassen, Aufzählungstypen

#### \* Vererbung mit Java-Schnittstellen

- Java-Schnittstellen, Arten der Vererbung
- Objektkomposition und Methodendelegierung
- Implementierung mehrerer Schnittstellen
- DAO-Designmuster

#### \* Generics und Collections

- Generische Klassen und Typparameter
- Typinferenz (Diamond)

### Voraussetzungen

Fundamentals of the Java Programming Language~4272

oder dem entsprechende Kenntnisse

### Hinweise

Version: 14

- Collections und Generics
- Schnittstellen List, Set und Map
- Stack und Deque
  
- 3. Tag
- \* Verarbeiten von Zeichenfolgen
- StringBuilder und StringBuffer
- Wesentliche Zeichenfolgenmethoden
- Parsen von Text in Java, Inputverarbeitung mit Scanner
- Textausgabe und Formatierung
- Reguläre Ausdrücke mit den Klassen Pattern und Matcher
  
- \* Exceptions und Assertions
- Exception-Kategorien, Java-Standardklassen von Exceptions
- Erstellen eigener Exception-Klassen
- Klauseln try-catch und finally
- try-with-resources und Schnittstelle AutoCloseable
- Klausel multi-catch
- Best Practices für Exceptions, Assertions
  
- \* I/O-Grundlagen
- I/O mit Java
- Lesen des Konsoleninput-Streams, Schreiben an die Konsole
- I/O-Streams, Verkettung von I/O-Streams, Kanal-I/O
- Lesen und Schreiben von Objekten mit Serialisierung
  
- \* Datei-I/O mit NIO
- Schnittstelle Path, Klasse Files
- Datei- und Verzeichnisvorgänge
- Verwalten von Dateisystemattributen
- Lesen, Schreiben und Erstellen von Dateien
- Verfolgen von Änderungen am Dateisystem
  
- 4. Tag
- \* Threading
- Planung von Betriebssystemaufgaben
- Erkennen von Multithread-Umgebungen
- Erstellen von Multithread-Lösungen
- Gemeinsame Datennutzung über Threads
- Synchronisierung und Deadlock
- Unveränderbare Objekte

