

Java Trainings

Bewährt. Flexibel. Zukunftsoffen.



Inhalt

Altbewährt und jung geblieben	4
Softwarequalität	5
Java Performance	6
Funktionale Programmierung und OO	7
JAVA Developer Qualifizierungsplaner.....	8
DevOps und Continuous Delivery	9
Cloud-Anwendungen	10
JAVA DevOps Qualifizierungsplaner	11
Seminarübersicht	12
Software-Architektur im Kontext von Microservices.....	14
Exzellente Trainer und professionelle Organisation	15
Web Development	16
Mobile und Internet of Things	17
JAVA EE Qualifizierungsplaner.....	18
Big und Fast Data	19
Java Security	20
Ihre Integrata AG	21
Für Sie vor Ort.....	22

Altbewährt und jung geliebt Mit Java ist weiterhin zu rechnen

Java ist eine gut eingeführte Technologie – ein globaler Standard, bewährt seit über 20 Jahren. Und doch gehört Java noch lange nicht zum alten Eisen. Java ist über die Jahre gereift und hat sich dabei als erstaunlich flexibel und zukunfts offen erwiesen. Auch wenn sich Java in Konkurrenz zu einer fortwährenden Diversifizierung der Angebote für Entwickler behaupten muss und nicht immer mit der Dynamik anderer Plattformen mithalten kann – der Senior unter den Programmiersprachen ist lebendiger denn je und bleibt eine Größe, mit der weiterhin zu rechnen ist. Es gibt keine andere Sprache, die es so einfach macht, produktiv zu arbeiten, und die so erfolgreich ist.

Was 1995 als Sprache für simple Browser-Oberflächen vorgestellt wurde, ist heute die mit Abstand am meisten benutzte und am weitesten verbreitete Programmiersprache weltweit: Java-Programme

laufen auf großen Server Clustern in der Cloud wie auch auf Desktops und Smartphones. Sogar Chip-Karten werden mit Java-Anwendungen ausgeliefert.

Die aktuelle Java-Version 8 basiert auf den etablierten Paradigmen der objekt-orientierten Programmierung, aber auch funktions- und aspektorientierte Ansätze werden unterstützt. Die mitgelieferte Standard-Klassenbibliothek ist enorm vielfältig. Sie ermöglicht unter anderem die parallelisierte Ausführung, bietet schnelle Streams zur Datenübertragung und implementiert Map-Reduce-Algorithmen zur Verarbeitung von Big Data.

Java ist mehr als nur Programmiersprache. Mit welcher Vielfalt an Möglichkeiten Java aufwarten kann, erschließt sich auf den folgenden Seiten. Entsprechend umfassend ist auch das Spektrum an Java-Schulungen, das die Integrata AG

bietet: Weiterbildung für Experten wie Java-Architekten, Software-Ingenieure, Java- und App-Entwickler sowie DevOps in den Bereichen

- Programmierung
- Architektur, Design, Modellierung
- Verfahren und Werkzeuge
- Frameworks und Produkte

Java zählt auch in Zukunft. Eigentlich wurde Java für den „M2M“-Bereich und kleinste Endgeräte geschrieben. Genau deshalb ist es nun das ideale Fundament für das heranbrechende Zeitalter des „Internet of Things“. Der Kreis schließt sich. Der Senior bleibt nicht nur gefragt, ihm scheint sogar ein zweiter Frühling zu blühen.

Es lohnt sich, bei Java auf dem Laufenden zu bleiben!



Softwarequalität

Alles im Griff – Techniken und Verfahren

Software muss wartungsfreundlich, flexibel, portierbar, wiederverwendbar, robust, agil und vieles mehr sein. Für eine hohe Codequalität sorgen Techniken und Verfahren wie:

- Ein durchdachtes **Projektkonzept**, für das ein Corporate Style entwickelt wurde, und das Qualitätskriterien, Regeln, die geplante Architektur sowie die verwendeten Techniken, Methoden und Modellierungsmittel klar definiert.
- Eine moderne komponenten- und schichtenbasierte **Architektur**, „Best Practice“-Techniken sowie der Einsatz moderner Frameworks.
- **Testings**, bei denen neben Integrationstests auch Unittests eine immer größere Rolle spielen. Entwickler sollten Testframeworks, Mocking und Tests von Legacy-Code beherrschen und mit komplexen Testfällen umgehen können.

- **Messungen** zur Sicherstellung der fachlichen und technischen Qualität.

Auch wenn der hohe initiale Aufwand beim konsequenten Einsatz der genannten Techniken abschreckt – die Vorteile überwiegen:

- Die Verfahren schärfen sich gegenseitig. Beispielsweise erfordern Unittests ein striktes Einhalten moderner Architekturregeln. In diesem Fall sind Probleme, die beim Testen auftreten, Symptome für Designfehler.
- Eine hohe Testabdeckung ist Voraussetzung für konsequentes Refactoring und erhält damit die Agilität und gleichzeitig die Robustheit der Software.
- Mit Hilfe von Messverfahren können Probleme hinsichtlich Kopplung, Kohäsion und Kapselung identifiziert werden.

Implementierungen können so leicht bewerkstelligt werden und Releasezyklen können eine höhere Frequenz bekommen.

- Durch Messungen lässt sich die Einhaltung von Unternehmensregeln wie Corporate Styles, Clean Code Regeln oder Konvention vor Dokumentation überprüfen.
- Regelmäßige Messungen decken Degenerationserscheinungen auf und man kann rechtzeitig eingreifen.
- Tools wie „Sonarcube“ können Ergebnisse und Statistiken graphisch aufbereiten und Kosten für die Behebung von Problemen berechnen.

Nur der flächige Einsatz aller qualitätsrelevanten Techniken und Verfahren unterstützt das Erreichen hoher Codequalität. Dabei ist eine Akzeptanz bei allen Entwicklern und Entscheidern wichtig und dafür wiederum ein Verständnis für die notwendigen Technologien und deren Vorteile im Alltag.

Ausgewählte Seminare zur Java Softwarequalität

Testgetriebene Entwicklung mit Java

Seminar-Nr. **33082**

Dauer: **3 Tage**

www.integrata.de/33082

Die Umsetzung agiler Methoden zur Softwareentwicklung verlangt eine umfangreiche Realisierung von Test-Programmen. Sie lernen in diesem Seminar die Test-Programmierung mit Java im Detail kennen. Sie konzipieren und realisieren Unit-, Integrations- und System-Tests und können diese effizient in

Suiten gruppieren. Weiterhin präsentieren wir Ihnen eine Reihe von alternativen Ansätzen zur Formulierung von Testfällen. Die Anbindung der Anwendungsprogrammierung an ein Ticket-System sowie die Integration in den kontinuierlichen Build-Prozess runden das Seminar ab.

Refactoring von Java-Programmen

Seminar-Nr. **33058**

Dauer: **2 Tage**

www.integrata.de/33058

Sie lernen in diesem Seminar erprobte Strategien zum Refactoring von Java-Programmen kennen. Kriterien, die die Notwendigkeit für einen Refactoring-Zyklus sowohl bei internen als auch bei extern realisierten Projekten signalisieren, werden vorgestellt. Dadurch erreichen Sie eine deutliche Steigerung der Qua-

lität Ihrer Unternehmens-Software und damit verbunden eine Sicherung Ihrer Investitionen. Im Rahmen der praktischen Übungen setzen Sie die gelernten Verfahren um und erhalten damit Referenzbeispiele für "schlechten" und "guten" Java Code.

Java Performance

Leistung zielgerichtet und passgenau optimieren

Zeit ist in unserer modernen Welt zu einem Faktor geworden, der über den Erfolg oder Misserfolg ganzer Geschäftsmodelle entscheiden kann. Wer das Tempo der Konkurrenz nicht mitgehen beziehungsweise übertreffen kann, ist schnell aus dem Rennen. Folgerichtig spielt Geschwindigkeit auch in der IT-Landschaft eine wichtige Rolle, aber nicht nur.

Neben der eigentlichen Performance-Steigerung zählen die Analyse von Code und das Monitoring von Anwendungen zu den Aufgaben, die mit leichtgewichtigen Anwendungen einhergehen.

Welche Performance- und Monitoring-Möglichkeiten bestehen?

- Stresstools wie z.B. Apache-Benchmark, Apache-JMeter, LOIC oder HOIC helfen, die Anwendungen und Server verschiedenen Szenarien auszusetzen.
- Für das Monitoring stehen uns neben Werkzeugen wie jps, jstat, jconsole, jvisualvm und jmc auch Access-Logfile, Error-Logfile und HEAP-Dump-Analysen zur Verfügung.
- Darüber hinaus müssen Entwickler die Arbeitsweise der Java Virtuellen Maschine verstehen. Sie müssen wissen, mit welchen Parametern sich diese konfigurieren lässt und wie beispielsweise das Zusammenspiel

mit den unterschiedlichen Speichersegmenten aussieht. Des Weiteren ist die richtige Wahl der verwendeten Objekte aus den zahlreichen Bibliotheken und Frameworks von entscheidender Bedeutung.

- Außerdem spielt die richtige und ausgewogene Konfiguration der Frameworks sowie der Serverlandschaft eine große Rolle.

Der Aufgabenstellung Java Performance zu verbessern kann man sich, wie sich zeigt, sowohl von architektonischer, administrativer, konfiguratorischer wie auch programmatischer Seite her nähern. All diese Maßnahmen tragen dazu bei, die drei limitierenden Faktoren CPU, Speicher und IO-Zugriffe zu optimieren.

Ausgewählte Seminare zu Java Performance

Optimierung und Tuning von Java-Programmen

Seminar-Nr. **3316**

Dauer: **5 Tage**

www.integrata.de/3316

Dieses Seminar wendet sich an erfahrene Java-Programmierer, die hier Anregungen zum Performance Tuning und zur Speicheroptimierung für Java-Anwendungen erhalten.

Neben den elementaren Regeln zur Optimierung werden zur Vertiefung Beispiele aus der Praxis vorgestellt, an denen die Machbarkeit der präsentierten Lösungsvorschläge aufgezeigt wird.

Verfügbarkeit, Performance und Betrieb von Java-Anwendungen und Applikationsservern

Seminar-Nr. **33073**

Dauer: **3 Tage**

www.integrata.de/33073

In diesem Seminar vermitteln wir Ihnen, welche Besonderheiten bei der Administration Java-basierter Server-Anwendungen (z.B. Applikationsserver wie Tomcat oder JBoss) beachtet werden müssen. Dazu lernen Sie detailliert die Arbeitsweise der Java Virtual Machine und dabei insbesondere die Multithreading-Fähigkeiten, die Speicherkonfiguration und

die Garbage Collection kennen. Die Methoden zur Erstellung und Analyse von JVM-Dumps sowie Werkzeuge zur Überwachung eines auch heterogenen JVM-Clusters sind Ihnen nach Besuch des Seminars bekannt. Besonderer Wert wird auch auf das Erkennen und Beheben von Fehlersituationen gelegt.

Funktionale Programmierung und OO

Leichter, schlanker, schneller

Mit Java hat sich in den letzten 20 Jahren das Paradigma der objektorientierten Programmierung als Standard etabliert. Alternativ dazu wurde – im akademischen Raum – am Paradigma der funktionalen Programmierung gearbeitet, mit Sprachen wie Lisp, ML und anderen.

Seit einigen Jahren entwickeln sich neue Sprachen (wie Scala und Ceylon), die versuchen, beide Paradigmen unter einem gemeinsamen Dach zusammenzufassen. Auch Java (wie auch C# und C++) wurde um funktionale Elemente erweitert. Welche Vorteile bringen diese Features den objektorientierten Sprachen?

- Die funktionale Programmierung favorisiert immutable Elemente. Diese lassen sich einfacher verstehen als änderbare Elemente und bedürfen keiner Synchronisation.
- Funktionen können als Werte behandelt werden und neben ihren eigenen Parametern auch Werte der Umgebung benutzen, in welcher sie erzeugt werden (Closures).
- Die Typen von Variablen und Funktionen können häufig automatisch vom Compiler berechnet werden.
- Funktionen können "anonym" definiert werden. Solche Funktionen werden als Lambdas bezeichnet.
- Funktionen können partiell ausgewertet werden.
- Objektorientierte Sprachen verwenden das Konzept der imperativen Programmierung. Sie beschreibt das „Wie“ der Lösung. Funktionale Sprachen favorisieren dagegen den deklarativen Stil, der das "Was" beschreibt.

Die Verwendung funktionaler Konzepte führt zu einer drastischen Reduktion von Code. In Java 8 etwa können anonyme Klassen durch Lambdas ersetzt werden - aus fünf Zeilen Code kann eine einzige werden. Der Quelltext wird so leichter lesbar und auch leichter wartbar, was wiederum die Produktivität der Entwickler steigert.

Für objektorientierte Softwareentwickler bedeutet das zwar eine radikale Umstellung auf ein komplett anderes Erscheinungsbild der neuen Features. Sobald ihnen jedoch die Vorteile, die die Verwendung funktionaler Elemente bieten, klar werden, möchten sie diese nicht mehr missen.

Ausgewählte Seminare zur Funktionalen Programmierung und OO

Java 8 / JDK8 Update

Seminar-Nr. **3300**

Dauer: **2 Tage**

www.integrata.de/3300

Das JDK 8 enthält insbesondere durch die Einführung der Elemente der funktionalen Programmierung eine ganze Reihe komplexerer Erweiterungen. Sie lernen die neuen Features

von Java 8 in diesem Seminar kompakt und zielorientiert kennen.

Scala

Grundlagen der Programmierung

Seminar-Nr. **33080**

Dauer: **3 Tage**

www.integrata.de/33080

Mit Scala steht Ihnen eine Programmiersprache zur Verfügung, die sich nahtlos in eine Java Laufzeitumgebung integriert. Sie lernen hier diese moderne Programmiersprache fundiert kennen. Sie erkennen die Vorzüge von Scala insbesondere im Bereich der objektori-

entierten Umsetzung und können die Einsatzmöglichkeiten von Scala in Ihren Projekten einschätzen.

JAVA Developer

Ihr persönlicher Qualifizierungsplaner



DevOps und Continuous Delivery

Softwareentwicklung in schnellen und agilen Zeiten

Softwareentwicklungen legen heutzutage ein rasantes Tempo hin. Qualitätssicherung wird hier zu einer großen Herausforderung: Continuous Delivery eröffnet die Möglichkeit, dass zu jeder Zeit eine aktuelle Version der erstellten Software auf eine Zielumgebung aufgebracht werden kann. Zudem kann eine Anwendung auf eine Umgebung deployed werden, zum Zweck der automatisierten und manuellen Tests. Sind diese Tests rein automatisch und die einzigen Abnahmekriterien, so kann auch das eigentliche Deployment auf die Produktionsumgebung beziehungsweise einen Teil davon automatisch erfolgen. Dann spricht man von Continuous Deployment.

Um diese hohe Geschwindigkeit zu ermöglichen, sind neben agilen Prozessen vor allem DevOps von essentieller Bedeutung.

DevOps ist die Verbindung von Development und IT Operations. Damit soll durch gemeinsame Anreize, Prozesse und Werkzeuge eine effektivere und effizientere Zusammenarbeit der Bereiche Entwicklung, IT-Betrieb und Qualitätssicherung ermöglicht werden. DevOps hat das Ziel, diese Bereiche in der Rolle des Buildmanagers/Build-Teams/DevOps Engineers konsequent zu verzahnen.

Dadurch können betriebsrelevante Fragen, die beim herkömmlichen, getrennten Ansatz erst zu einem späten Stadium auftauchen und das eigentliche Deployment blockieren können, frühzeitig erkannt, gestellt und beantwortet werden. Auf diese Weise wird der Betrieb von Anfang an in die Entwicklung eingebunden. Die Software kann so früh wie möglich als sogenanntes "Walking Skeleton" deployed werden: Ein sicheres Mittel, den Worst Case, nämlich dass sich Fehler erst beim Livegang zeigen, auszuschließen.

Gute DevOps-Teams sind sowohl mit entwicklungsnahe Buildmanagern als auch mit betriebsnahe Administratoren besetzt. Sie begleiten den gesamten Prozess von den ersten Ansätzen bis zur Fertigstellung und können so jederzeit schnell auf neue Anforderungen beider Seiten reagieren.

Ausgewählte Seminare zu DevOps und Continuous Delivery

Java Build-Management

Seminar-Nr. **33067**

Dauer: **3 Tage**

www.integrata.de/33067

Eine stabile Build-Umgebung ist essenziell, um in den einzelnen Phasen eines Softwareentwicklungsprojektes den Überblick über den Entwicklungsstand zu behalten. In diesem Seminar lernen Sie, wie Sie interne und externe Abhängigkeiten verwalten und

kontinuierlich den Status Ihres Projektes überwachen können. Nach diesem Seminar kennen Sie praxisbewährte Verfahren und Werkzeuge, um eine Build-Umgebung für Java-Projekte planen, erstellen und pflegen zu können.

Test und Qualitätssicherung in Java- und Java EE-Projekten

Seminar-Nr. **33065**

Dauer: **3 Tage**

www.integrata.de/33065

Dieses Seminar zeigt Ihnen etablierte Werkzeuge und Verfahren zum komplexen Testen von Java-Applikationen. Dabei werden die besonderen Bedingungen und Anforderungen der Qualitätssicherung bei objektorientierter Umsetzung herausgearbeitet. Anhand konkre-

ter Beispiele wird die praktische Umsetzung der vermittelten Inhalte aufgezeigt.

Cloud-Anwendungen

Flexible Services mit Java und Docker

Die Entwicklung von Anwendungen für die Cloud bietet neue Chancen und attraktive Geschäftsmodelle. Im Gegensatz zu herkömmlichen On-Premise-Anwendungen skalieren Cloud-Anwendungen deutlich besser. Und sie sind kosteneffizienter. Aufgrund der dezentralen Verteilung sind sie außerdem häufig performanter. Allerdings ist eine Cloud-Anwendung weit mehr als eine virtuelle Maschine, die einfach in der Cloud läuft. Um die Fähigkeiten der Cloud richtig zu nutzen, müssen die Software und der Entwicklungsprozess darauf zugeschnitten sein.

Cloud-Anwendungen verfolgen einen Microservices Ansatz. Das bedeutet, dass die große schwergewichtige Anwendung, die oftmals das Markenzeichen herkömmlicher JavaEE Anwendungen ist, in viele kleine, autarke Dienste zerlegt wird. Diese kommunizieren über einfache Protokolle wie MQ oder REST miteinander. Das erleichtert neben der Entwicklung auch die Wartung und Updates ihrer Applikation.

Durch Container-Dienste wie Docker kann die Anwendung einmal gebaut und unverändert auf alle Umgebungen verteilt werden.

Einerseits erleichtert und beschleunigt dies das Deployment, also die Bereitstellung der Anwendungen. Die Container, die alle nötigen Pakete enthalten, können problemlos als Dateien transportiert und installiert werden. Andererseits stellt dies aber auch andere Anforderungen an die Entwicklung, die in diesem Fall mehr mit dem klassischen Betrieb verzahnt wird. Der Fokus liegt dabei auf der Prozessverbesserung im Hinblick auf die Qualität der Software, die Geschwindigkeit der Entwicklung und Auslieferung sowie das Miteinander der beteiligten Teams (DevOps-Ansatz).

Ausgewählte Seminare zu Java Cloud-Anwendungen

RESTful Web Services
Implementierung mit Java

Seminar-Nr. **33049**

Dauer: **2 Tage**

www.integrata.de/33049

Dieses Seminar präsentiert Ihnen sowohl die Architektur-Prinzipien als auch die konkrete Implementierung von RESTful Web Services. Sie verstehen, warum dieser Ansatz in den

letzten Jahren so populär geworden ist und lernen, REST in Ihren Projekten konkret einzusetzen.

Docker und Java Docker-Container für Java-Applikationen

Seminar-Nr. **33032**

Dauer: **3 Tage**

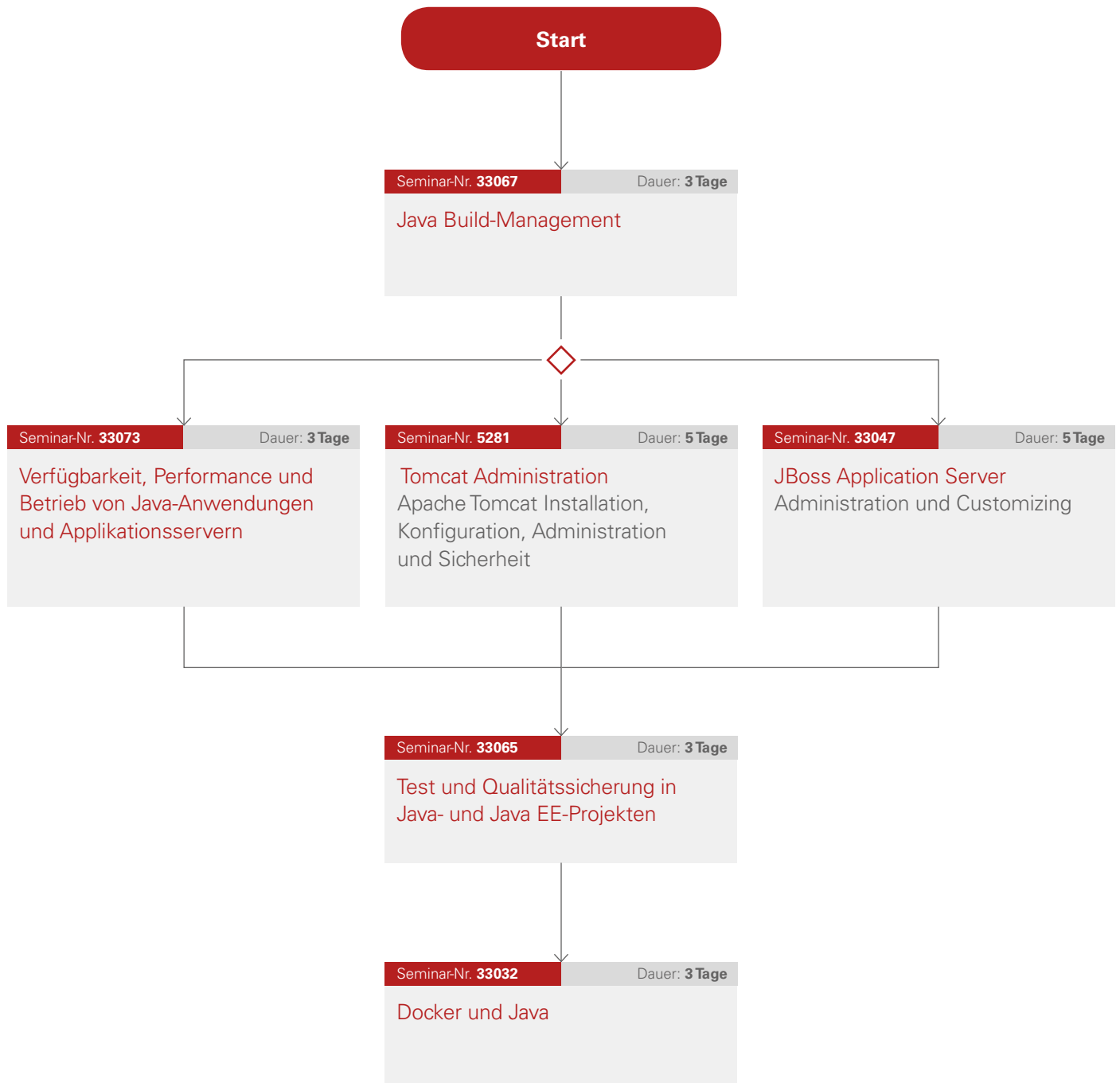
www.integrata.de/33032

Docker-Container werden immer häufiger in modernen System-Architekturen eingesetzt. Wir vermitteln Ihnen in diesem Seminar die Grundlagen von Docker. Sie lernen, Docker-

Images für Java-Anwendungen zu konfigurieren und damit komplexe Anwendungen umzusetzen.

JAVA DevOps

Ihr persönlicher Qualifizierungsplaner



Seminarübersicht

Java

Architektur, Design, Modellierung		Dauer	Java Programmierung		Dauer
UML			Überblick		
33063	UML für Java-Programmierer	3	3301	Java und Java EE - Überblick für Führungskräfte	2
Design Patterns			Basics, Advanced, Experts		
3310	Design Pattern in Java-Programmen	3	3302	Java Grundlagen	5
33058	Refactoring von Java-Programmen	2	3304	Java Erweiterungen I	5
33031	JEE Patterns	4	3323	Java Erweiterungen II	3
Enterprise Architekturen			3324	Java Intensivtraining für Sprachumsteiger mit Erfahrung in OOP	5
33038	Enterprise Architekturen mit Java und der Java Enterprise Edition	4	33082	Testgetriebene Entwicklung mit Java	3
33060	Enterprise Service Bus	4	3316	Optimierung und Tuning von Java-Programmen	5
33125	Microservices	2	3329	OSGi	2
2054	Service Oriented Architecture (SOA) Kompakt - Services, Domänen und Bebauungspläne	2	3300	Java 8 / JDK8 Update	2
Big und Fast Data			3338	Java Batch - Job-Programmierung mit Spring Batch und JSR 352	2
33090	NoSQL: Nicht-relationale Datenbanken	4	Java Enterprise Edition		
33034	Apache Hadoop Anwendungsentwicklung	3	3313	Einführung in die Java Enterprise Edition (Java EE)	3
			632	Powertraining: Java Enterprise (JEE)	6
			33013	Anwendungsentwicklung mit JEE7	3
			Data Access		
			33042	JPA - Das Java Persistence API	4
			Web Frontends		
			33077	JavaServer Faces 2	4
			3307	Java Servlets und JavaServer Pages	5
			CDI		
			33057	Enterprise JavaBeans 3	3
			SOAP und REST		
			33033	Web Services mit Java	3
			33049	RESTful Web Services	2
			Security		
			33059	Java EE- und SOA-Security	3
			5213	Web Hacking	3
			Desktop, Mobile und Kleingeräte		
			3308	GUI-Entwicklung mit Java Swing und JFC	3
			33074	JavaFX - Rich Internet Applications mit JavaFX	3
			33102	Mobile Web Applications	3
			33112	Java für Kleingeräte	2

Verfahren und Werkzeuge		Dauer
Build und Releasemanagement		
33067	Java Build-Management	3
Testen & QS		
33065	Test und Qualitätssicherung in Java- und Java EE-Projekten	3
DevOps		
33032	Docker und Java - Docker-Container für Java-Applikationen	3
Source Code Management		
33096	GIT - Ein Versionsverwaltungssystem	2
33062	Subversion	2
Team Collaboration		
33086	Java Team Collaboration und Dokumentation in Java Projekten	2
Serverüberwachung & Administration		
33073	Verfügbarkeit, Performance und Betrieb von Java-Anwendungen und Applikationsservern	3

Frameworks und Produkte		Dauer
Spring		
33045	Spring - Einführung in die Programmierung leichtgewichtiger Java-Anwendungen	4
33089	Spring Aufbau	3
33088	Webanwendungen mit Spring MVC, JSF und Spring Web Flow	4
Eclipse		
33036	Eclipse IDE für Java-Entwickler	3
33087	Eclipse Rich Client Platform	5
Android		
33110	Entwickeln von Android Apps	5
33111	Entwickeln von Android Apps für Java-Entwickler	3
Frameworks		
33039	Open Source Frameworks und Tools für Java und Java EE	3
33099	Vaadin - Ein Web Framework	2
33076	Wicket - Webapplikationen mit HTML und Standard-Java	3
Applikationsserver		
5281	Tomcat Administration	5
33047	JBoss Application Server	5
4585	WebSphere Application Server	5
34008	JBoss Enterprise Application Plattform (EAP) 7 / Wildfly 10 Administration	4
Weitere Produkte und Sprachen		
33126	Apache Camel - Ein Open Source-Integration Framework	3
33085	Apache ActiveMQ	3
33079	Liferay Portal Server	3
33078	Alfresco	3
33080	Scala - Grundlagen der Programmierung	3

Software-Architektur im Kontext von Microservices

Kleine Dienste – große Wirkung

Derzeit ist kein Monolith vor ihnen in Sicherheit. Die IT hat sich mal wieder neu erfunden und treibt ein neues Schlagwort durch das Dorf. Um künftig Skalierbarkeit sicherzustellen und die Wartungskosten im Griff zu haben, werden große Applikationen in kleine Dienste zerlegt. Was man erhält bezeichnet man als Microservicearchitektur. Ein Microservice ist demnach ein unabhängig lauffähiges Artefakt, das dem Gesamtsystem eine kleine oder sehr kleine Menge an Verantwortlichkeiten bereitstellt.

Doch wie gut funktioniert die Implementierung eines Systems aus Microservices tatsächlich? Meist folgt nach der ersten Aufbruchstimmung die Ernüchterung und man sieht sich mit zahlreichen Fragen konfrontiert:

- Wie schneide ich meine Services?
- Welche Art von Interface biete ich eigentlich nach außen hin an?
- Wie transportiere ich Zustände, die vorher im Monolith einfach verwaltbar waren (Security, Globale Filterparameter)?
- Darf ich Ergebnisse anderer Services cachieren?
- Wie kann ich eine produktionsnahe Integrationsumgebung aufbauen und pflegen?
- Wie finden sich die Services untereinander?
- Orchestriere ich die Services im Browser oder ist eine Fassade besser?
- Wie gehe ich mit Schnittstellenänderungen um, ohne meine Abhängigkeiten alle konkret zu kennen?
- Und nicht zuletzt organisatorische Fragen wie beispielsweise: „Wie viele Services darf ein Team haben?“

Leider lässt sich nicht alles sofort beantworten. Und auch wenn im Java Umfeld durch Spring Boot sehr viel technische Unterstützung für die Umsetzung von Microservices vorhanden ist - alles kann damit nicht abgedeckt werden. Umso wichtiger ist es, ein Problembewusstsein für die organisatorischen, architektonischen Fragen zu entwickeln. Wer sensibilisiert ist, wird Herausforderungen beim Aufbau eines Systems von lose gekoppelten Services schneller erkennen und kann mögliche Lösungen zielführend diskutieren.

Ausgewählte Seminare zu Software-Architektur

Microservices

Seminar-Nr. **33125**

Dauer: **2 Tage**

www.integrata.de/33125

In diesem Seminar erhalten Sie eine anerkannte Charakterisierung von Microservices und können damit Vor- und Nachteile dieses Ansatzes bewerten. Sie lernen, Anwendungen auf Basis von Microservices zu modellieren

und umzusetzen. Weiterhin erhalten Sie konkrete Richtlinien und Regeln, die bei der Einführung von Microservices im Softwareentwicklungsprozess sowie im Produktionsbetrieb zu beachten sind.

Enterprise Architekturen mit Java und der Java Enterprise Edition

Seminar-Nr. **33038**

Dauer: **4 Tage**

www.integrata.de/33038

Sie modellieren in diesem Seminar die Architektur und das Design einer komplexen Anwendung. Aus den Anforderungen für eine verteilte Anwendung werden verschiedene Basis-Designs abgeleitet, die ohne Änderung für typische Szenarien verwendet werden können.

Dabei werden auch leichtgewichtige Modelle berücksichtigt. Die vermittelten Inhalte werden anhand einer Referenzanwendung demonstriert.

Exzellente Trainer und professionelle Organisation

Perfekte Basis für Ihren Lernerfolg

In einem technologisch und konzeptuell so anspruchsvollen Gebiet wie Java sind motivierte und engagierte Seminarleiter besonders wichtig. Wir haben einen Stamm an hochqualifizierten und praxiserfahrenen Trainern, die seit vielen Jahren mit uns zusammenarbeiten. Sie schätzen uns als verlässlichen und kompetenten Partner. Beispielsweise entlasten wir sie von allen organisatorischen Aufgaben und kümmern uns darum, dass alles reibungslos läuft. Davon profitieren auch unsere Kunden:

- Installation und Ausstattung in den Seminarräumen sind perfekt organisiert. Alle Rechner werden vor jedem Seminar mit allen benötigten Werkzeugen komplett neu aufgesetzt und funktionieren. Der Trainer kann sich damit von Anfang an uneingeschränkt auf sein Seminar konzentrieren.
- Für jedes Seminar existieren ein fertiger Satz von Unterlagen, ein Praktikum mit Muster-Beispielen und ein Leitfaden. Damit ist für die Durchfüh-

rung eine solide Basis geschaffen, die der Trainer individuell an die Teilnehmer-Voraussetzungen und -Anforderungen anpassen wird.

- Seminarunterlagen werden permanent aktualisiert und angepasst. Ebenso werden im Java-Umfeld regelmäßig neue Themen angeboten. Interessierte Trainer können an den Überarbeitungen beziehungsweise der Neuerstellung aktiv mitwirken und eigene Ideen einbringen.



Web Development

Ein Java-Klassiker bleibt aktuell

Browser-Anwendungen wurden ursprünglich nach dem MVC2-Pattern unter Benutzung einer serverseitigen View konzipiert. Heutzutage überwiegen Single Page-Applikationen mit asynchronen AJAX-Aufrufen unter konsequenter Verwendung einer RESTful Architektur. Das führte zu einer Vielzahl an neuen Frameworks und Sprachen. Doch trotz des grundsätzlichen Wandels sind beide Herangehensweisen weiterhin im Spiel.

MVC2-Anwendungen werden mit JavaServer Faces, einem Bestandteil der Java Enterprise Edition, oder auch dem immer noch sehr beliebten proprietären Spring MVC-Framework implementiert. Beide Frameworks sind ausgereift und bieten ein einfaches Programmiermodell mit mächtigem Funktionsumfang.

RESTful Server werden mit JAX-RS programmiert. Diese Bibliothek ist zwar deutlich jünger als JSF, aber auch wesentlich schlanker; da REST bereits durch den http-Standard umgesetzt ist.

In Konkurrenz zu JavaScript-Frameworks wie jQuery, Angular, Backbone und React, um nur einige der Neuentwicklungen zu nennen, verlor Java seine Führungsposition. Mit Hilfe eines Tricks können Browser-Anwendungen dennoch auch weiterhin mit Java programmiert werden: Ein Source-to-Source-Compiler transcompiliert bzw. transpiliiert Java-Logik nach JavaScript:

JavaServer Faces definiert AJAX-Aufrufe deklarativ als Bestandteil der Seitendefinition – eine direkte Browser-Programmierung ist unnötig.

Frameworks wie Google Web Toolkit (GWT) oder Vaadin definieren die Client-Oberfläche und den RESTful Server mit Java-Programmen. Der Client-Code wird direkt nach JavaScript transpiliiert. Eclipse RCP- und JavaFX-Anwendungen können nach moderaten Anpassungen im Browser ausgeführt werden.

Mit ECMA2016 und Java 8 ähneln sich die beiden Programmiersprachen immer mehr. Eine standardisierte Übersetzung von Java direkt nach JavaScript rückt damit in greifbare Nähe.

Ausgewählte Seminare zu Java Web Development

JavaServer Faces 2 Das Standard User-Interface-Framework für Java EE-Anwendungen ab Version 6

Seminar-Nr. **33077**

Dauer: **4 Tage**

www.integrata.de/33077

Nach diesem Seminar kennen Sie die Architektur des JSF 2 Frameworks. Anhand einer Beispielanwendung werden die Themen Managed-Beans und CDI, Navigation, Konvertierung und Validierung bis zum Event-Handling detailliert besprochen. Die Besprechung

einiger populärer Komponenten-Bibliotheken (Primefaces, RichFaces, IceFaces) rundet das Seminar ab.

Mobile Web Applications Web Anwendungen für iPhone, Android, Blackberry und Windows Phone

Seminar-Nr. **33102**

Dauer: **3 Tage**

www.integrata.de/33102

Sie lernen in diesem Seminar die besonderen Anforderungen der Programmierung von Web Anwendungen für mobile Endgeräte kennen. Wir zeigen Ihnen, wie Sie mit Hilfe von HTML5, CSS und JavaScript gefällige und benutzerfreundliche Web Seiten für Smartphones und Tablets erstellen und optimieren

können. Damit sind Sie in der Lage, einheitliche Anwendungen für die verschiedenen Plattformen (Android, iPhone, Blackberry, Windows Phone) und Devices (Smartphones, Tablets) zu erstellen.

Mobile und Internet of Things

Java verbindet die „Dinge“ der Zukunft

Aktuell ist Java weltweit auf mehr als 15 Milliarden Geräten installiert. Für das Jahr 2020 werden etwa 25 bis 50 Milliarden Java Devices prognostiziert. Ist das überhaupt realistisch?

Java ist zwar auf dem Server die dominante Plattform, allerdings erreicht die Anzahl von Servern weltweit bei weitem nicht diese Höhe.

Java ist auch auf dem Desktop durchaus präsent, hat sich aber gegenüber nativen Applikationen und Browser-basierten Anwendungen nicht entscheidend durchsetzen können. Außerdem schrumpft dieser Markt tendenziell.

Die Erklärung für die immensen Steigerungsraten finden wir an ganz anderer Stelle: Zum Beispiel im Markt für mobile Endgeräte in Form persönlicher Smartphones oder Tablets. Java konnte davon profitieren, dass Google mit seinem dominanten Betriebssystem Android auf Java-basierte Apps setzt.

Weitere Zuwächse verspricht der Bereich der Wearables wie Smartwatches und anderes mehr.

Auch moderne Geräte der Unterhaltungselektronik, TV-Boxen und Media-Player werden häufig mit Android ausgeliefert, auf jedem Blu-Ray-Player läuft Java.

Den größten Markt für Java-Endgeräte stellen jedoch Smart Cards und Embedded Devices dar! Zusammen mit den

Herstellern hat Oracle hierfür Java-Spezifikationen definiert: Java Embedded, die Micro Edition und Java Card. Damit programmieren und installieren Kartenhersteller bereits seit Jahren individuell angepasste Software auf SIM- und Kreditkarten.

Mit dem "Internet of Things" wird Java noch wichtiger. Java-basierte Kleingeräte werden dafür sorgen, dass praktisch alle „Dinge“ – egal ob Körpersensor, Küchenmaschine, Automotor oder Haus-elektronik – Daten erfassen und Aktionen ferngesteuert ausführen können. Die Java-Plattform ist für Hersteller die Technologie der Wahl: etabliert, ausgereift und sofort einsetzbar.

Ausgewählte Seminare zu Java Mobile und Internet of Things

Entwickeln von Android Apps für Java-Entwickler

Seminar-Nr. **33111**

Dauer: **3 Tage**

www.integrata.de/33111

Sie lernen in diesem Seminar die Programmierung von Applikationen für die Android-Plattform. Ihre Anwendungen sind somit aufgrund der Standardisierung kompatibel zu allen

Android-fähigen Endgeräten, insbesondere Mobiltelefonen, aber auch den kommenden Produkten im Kleingerätemarkt.

Java für Kleingeräte

Seminar-Nr. **33112**

Dauer: **2 Tage**

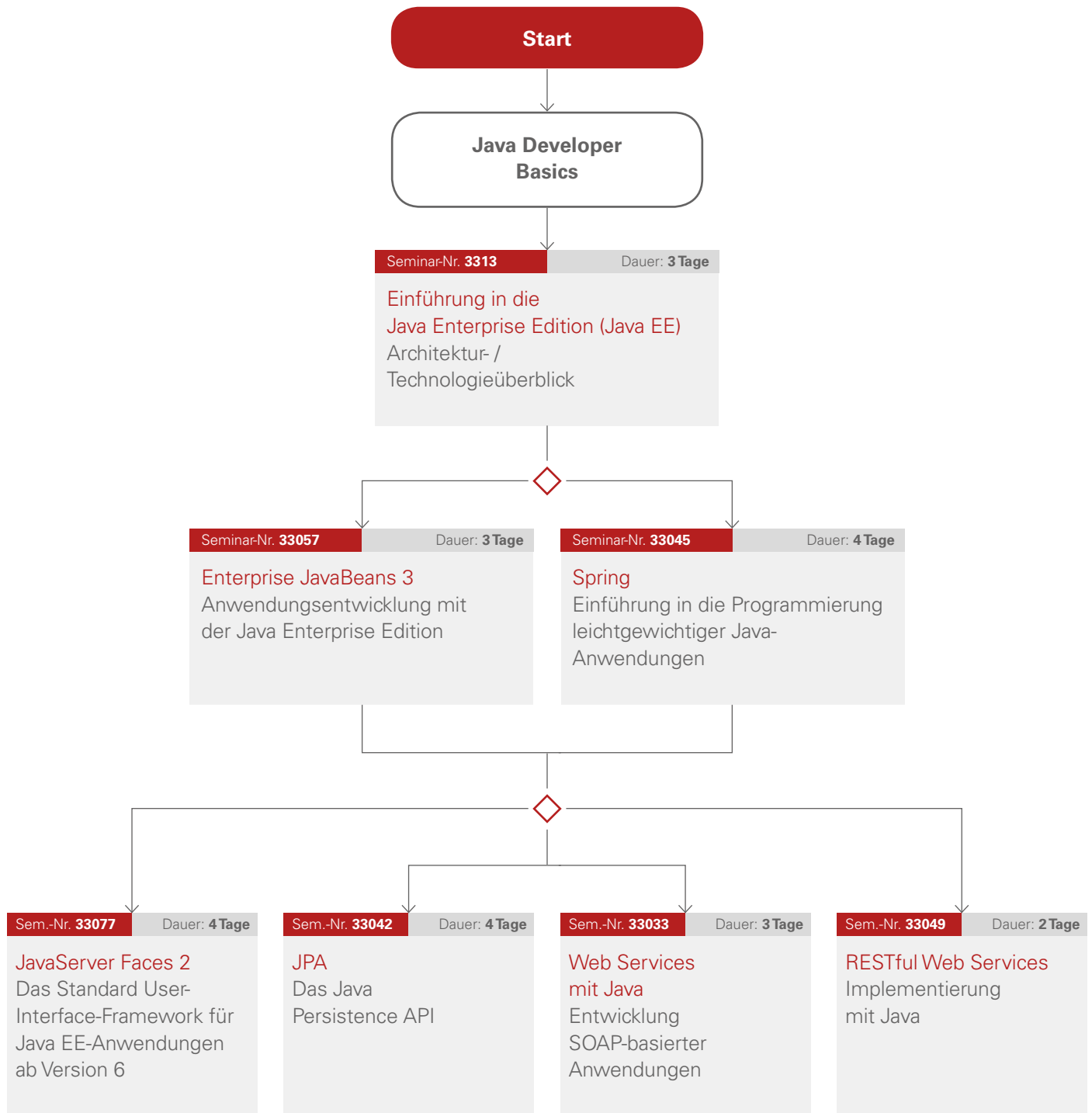
www.integrata.de/33112

Wir präsentieren Ihnen in diesem Seminar die verschiedenen Java-Spezifikationen für Kleingeräte: Java Card, Java Micro Edition mit Java TV und Embedded Java. Die für Sie relevanten

Technologien werden Ihnen angepasst an Ihre Bedürfnisse und Vorgaben fundiert und umfassend vermittelt.

JAVA EE Entwickler

Ihr persönlicher Qualifizierungsplaner



Big und Fast Data

Java-Lösungen für große Herausforderungen

„Big Data“ löste einen immensen Innovationsschub in der Informationsverarbeitung aus! Auf NoSQL-Prinzipien basierende Datenmodellierungen und darauf operierenden Verfahren wie Map-Reduce oder Graphen-basierte Analysen sind mittlerweile integraler Bestandteil moderner Software-Architekturen. Dabei hat sich der Fokus von „Big Data“ immer mehr in Richtung „Fast Data“ verschoben. Echtzeit-Analysen komplexer Daten sind quasi Standard – eine Anforderung, die im Hinblick auf das hereinbrechende Zeitalter des „Internet of Things“ für zunehmend größere Datenmengen erfüllt und immer schneller bewältigt werden muss.

Und die Menge an Daten, die gesammelt, aufbereitet und analysiert werden sollen werden im im Zeitalter des „Internet of Things“ weiter drastisch ansteigen.

Der Trend zu „Big and Fast Data“ beeinflusst auch die Java-Plattform: So wurden neue APIs zur Stream-orientierten, funktionalen und asynchronen Datenverarbeitung eingeführt. Alternativ konzipierte Sprachen wie Scala sind bei einer dynamischen und schema-orientierten Datenverarbeitung Java überlegen, erzeugen aber kompatiblen und interoperablen Bytecode. Und die Java Virtual Machine überzeugt mit einem neuen Garbage Collector sowie einer deutlich verbesserten Unterstützung von Native Memory: Programmierer wie Administratoren stehen vor einer Vielfalt an neuen Möglichkeiten und Herausforderungen.

Denn Java ist und bleibt für die meisten Unternehmen und Softwarehersteller eine gesetzte Strategie. Deshalb sind Java-Lösungen auch bei „Big Data“-Systemen weit verbreitet wie diese knappe aber repräsentative Auswahl zeigt:

- Apache bietet mit Cassandra und der HBase Key- bzw. Column-orientierte Datenbanken aus dem NoSQL-Umfeld.
- Eine Vielzahl von Analyse- und Reporting-Lösungen sind als klassische Java Enterprise Edition umgesetzt.
- Hadoop ist ein Apache Framework für skalierbare und verteilte Anwendungen und bildet damit das Rückgrat für parallel ausführbare Massendatenverarbeitung.
- Neo4J ist der Marktführer bei den graphenorientierten Datenbanken.

Ausgewählte Java Seminare zu Big und Fast Data

NoSQL: Nicht-relationale Datenbanken Technologien, Einsatzbereiche, Produktübersicht

Seminar-Nr. **33090**

Dauer: **4 Tage**

www.integrata.de/33090

In diesem Seminar werden Ihnen Architektur und Arbeitsweise von nicht-relationalen Datenbanken präsentiert. Dabei werden Ihnen die gängigen Kategorien (Dokumenten- und Graphen-orientiert, Key-Value) sowie darauf basierende Produkte im Einsatz gezeigt.

Sie lernen konkret, wie Daten innerhalb der Datenbanken strukturiert organisiert und abgelegt werden können. Daneben werden Ihnen effiziente Query-Strategien vermittelt.

Apache Hadoop Anwendungsentwicklung

Seminar-Nr. **33034**

Dauer: **3 Tage**

www.integrata.de/33034

Nach dem Besuch dieses Seminars sind Sie in der Lage, Anwendungen für Hadoop zu entwickeln, zu testen und zu debuggen. Sie verstehen die Architektur dieses Frameworks

und können damit effiziente Algorithmen zur Umsetzung komplexer Anforderungen implementieren.

Java Security

Schutz statt Risiko

Angriffe auf Daten und Spionage können Unternehmen in den Ruin treiben. Das Thema nahm in den letzten Jahren an Brisanz zu. Der Schutz nach außen wie nach innen wird immer schwieriger. Betroffen ist nicht nur die allgemeine Sicherheitslage, sondern insbesondere die IT-Security. Grund genug, sich mit der Sicherheit von Java-Anwendungen zu beschäftigen – insbesondere auch im Hinblick auf das "Internet of Things".

Welche Sicherheitsrisiken bestehen?

- Injection-Schwachstellen erlauben es, Code-Fragment in Abfragen einzubetten, und dann mit diesen SQL- oder LDAP-Abfragen zu manipulieren.
- Über Injection-Schwachstellen können auch die Authentifizierung und das Session-Management gekapert werden, um damit Passwörter, Session-IDs oder Benutzer-Identitäten zu über- beziehungsweise anzunehmen
- XSS-Schwachstellen ermöglichen die Implementierung von Schadsoftware in einen Browser. So können Benutzersitzungen übernommen, Seiteninhalte verändert oder der Benutzer auf böartige Seiten umgeleitet werden.
- Über unsichere, direkte Objektreferenzen können schwach erstellte oder schwach konfigurierte Anwendungen attackiert werden. Ziel ist, den Zugriff auf Daten, Dateien, Ordner oder Datenbankschlüssel zu erlangen.

- Sicherheitsrelevante Fehlkonfigurationen werden zum Risiko für Anwendungen, Frameworks, Applikations-, Web- und Datenbankserver sowie deren Plattformen.
- Beim Verzicht auf Verschlüsselung droht der Verlust von vertraulichen Daten.

Nicht nur Kriminelle nutzen solche Sicherheitslücken aus. Ein nicht zu unterschätzender Teil fällt auch in den Bereich der Sabotage und Spionage durch regierungsnahen Organisationen.

Gegen diese Sicherheitsrisiken kann man sich einerseits durch organisatorische Maßnahmen schützen, andererseits bietet eine entsprechende Konfiguration und Programmierung zahlreiche Lösungen.

Ausgewählte Seminare zu Java Security

Java EE- und SOA-Security

Schutz vor Hacker-Angriffen

Seminar-Nr. **33059**

Dauer: **3 Tage**

www.integrata.de/33059

Dieses Seminar wendet sich an Administratoren und Entwickler, die ihre Java Enterprise- und SOA-Anwendungen gegen interne und externe Hackerangriffe schützen wollen. Die Kenntnis der Angriffstechniken ist die Grundvoraussetzung für die erfolgreiche Abwehr von

Hackern. Deshalb liegt der Schwerpunkt des Seminars nicht auf theoretischen Betrachtungen, sondern es werden Sicherheits- und Abwehrstrategien sowie deren Umsetzung an konkreten Beispielen im Java-Umfeld herausgearbeitet.

Web Hacking

Angriffe gegen Webserver und -anwendungen erkennen und vorbeugen

Seminar-Nr. **5213**

Dauer: **3 Tage**

www.integrata.de/5213

Das Seminar ist so ausgelegt, dass es jedem, auch nicht technisch versierten, einen möglichst einfachen Einstieg in die Hacking Thematik bietet. Anhand vieler praktischer Beispiele und Erfahrungsberichte wird das notwendige Know-how vermittelt und notwendige Tools vorgestellt, um Vorgehensweisen von Hackern

zu verstehen und direkt erste Sicherheitskonzepte umsetzen zu können.

Das Seminar geht dabei auf alle wesentlichen Komponenten ein, die im Zusammenspiel mit Webanwendungen verbunden sind (Netzwerk-, System-/Server- und Anwendungssicherheit).

Ihre Integrata AG

inspire

Wir geben bereichernde
Entwicklungsimpulse
und schaffen Freiräume.



qualify

Wir stärken Menschen
in ihren sozialen und
fachlichen Kompetenzen.



change

Wir beraten und
unterstützen Sie im
Prozess des Wandels.



Für Sie vor Ort

in Deutschland



Hamburg • Hannover • Berlin • Münster • Manching
Dresden • Düsseldorf • Köln • Witten • Frankfurt/Main
Karlsruhe • Nürnberg • Stuttgart • München • Ravensburg

50 Jahre Erfahrung

200 festangestellte Mitarbeiter

80.000 Teilnehmer pro Jahr

1.400 erfahrene Referenten

1.200 Kernthemen

150 Berater

Durchschnittsnote **1,5**

90 Jahre am Markt

In **50** Ländern vertreten

200 Mio. Umsatz

4.500 internationale Referenten

1 eigenes E-Learning Produktionszentrum in Portugal

Weltweit **1.200** Mitarbeiter

300.000 Teilnehmer pro Jahr

mit der Cegos Group in Europa führend auf 5 Kontinenten aktiv



Europa • Osteuropa • USA • Kanada • Südamerika
Russland • Nahost / Emirate • Südafrika
China • Südostasien • Indien • Australien

Integrata AG
Zettachring 4
70567 Stuttgart

www.integrata.de

Seminaranmeldung

E-Mail: seminaranmeldung@integrata.de

Telefon: +49 711 62010 - 100

Telefax: +49 711 62010 - 267

Adressänderung · Kataloganforderung

E-Mail: infoservice@integrata.de

Telefon: +49 711 62010 - 300

Telefax: +49 711 62010 - 216

Impressum

Herausgeber: Integrata AG, Zettachring 4, 70567 Stuttgart

Sitz der Gesellschaft Stuttgart, Registergericht: Amtsgericht Stuttgart HRB 721012

Vorstand Ingmar J. Rath (Vorsitz), Dr. Andreas Dahmen, Hartmut Jöhnk

Vorsitzender des Aufsichtsrates Rechtsanwalt und Notar Stephan Haack

Verantwortlich für den Inhalt: Ingmar J. Rath, Dr. Andreas Dahmen, Hartmut Jöhnk

Konzeption: Kai Nellinger, Bozica Klein • Text: Enis Akmut, M.A.S. - Texte und mehr • Gestaltung: Die Designerei, Stuttgart

Fotos: Shutterstock (S.1 racorn, S.4 ESB Professional S.15 Wavebreakmedia), Getty (S.21 Morsa Images) • Druck: logo Print GmbH