

Skapa XML Web services med Java - 4 dagar

kurser 577

Du får lära dig att

- Utveckla, driftsätta och övervaka Web services och Web service-klienter med JAX-WS
- Implementera tjänsteorienterad arkitektur (SOA) med Web services
- Skapa och driftsätta WSDL-first och code-first Web services
- Bygga synkrona och asynkrona Web service-klienter i Java
- Leverera RESTful Web services för AJAX på serversidan
- Säkra Web services programmatiskt och deklarativt

Sammanfattning

Web services revolutionerar sättet på vilket företag kan kommunicera med varandra genom att göra applikationer på skilda maskinvaru- och programvaruplattformar interoperabla. Javas API:er för XML levererar en uppsättning kraftfulla verktyg för att utveckla tjänster. Du lär dig att utforma och skapa Web services med Java. Du utvecklar tjänster och klienter med de senaste, standardbaserade metoderna samt inför Web services som integrerar beprövade säkerhetsstrategier.

Vem bör delta

Programmerare, arkitekter, chefer och alla som vill integrera applikationer över webben. Kunskaper motsvarande 471, "Java-programmering: omfattande introduktion", eller motsvarande kunskaper förutsätts.

Praktiska övningar

Övningarna ger praktisk erfarenhet av att bygga Web services med Java och omfattar:

- Skriva en code-first Web service
- Binda komplexa XML-typer till Java-böner
- Skriva och driftsätta en WSDL
- Skapa en contract-first Web service från WSDL
- Bygga asynkrona Web service-klienter
- Kontrollera inventering från en webbläsare
- Autentisera och auktorisera åtkomst till Web service

Skapa XML Web services med Java - 4 dagar

kurser 577

Översikt över Web Services

Interoperabla applikationer med tjänsteorienterad arkitektur (SOA)

- Designa en SOA-integrationsarkitektur
- Utvärdera alternativ till SOA

Implementera SOA med Web services

- Kärntekniker: HTTP, XML, SOAP, WSDL
- Vad SOA inte har

XML-bearbetning i Java

XML:s grunder

- XML-syntax och namnutrymmen
- Beskriva XML med schema

Interagera med XML från Java

- Skapa och "parsa" XML med DOM
- Ordna och splittra med JAXB
- Anpassa XML till Java-bindningar

Definiera SOAP-meddelanden med WSDL

SOAP-meddelandens struktur

- SOAP:s betydelse för Web services
- Operationer, meddelanden och fel

Anatomin i ett WSDL-dokument

- Definiera gränssnitten i en Web service
- Specificera implementeringen
- Driftsätta WSDL

Generera WSDL-first Web Services

Bygga en Web Service

- Designa en tjänsteslutpunkt
- Specificera utbyte av meddelandeprotokoll
- Bevara flexibilitet och tøjbarhet

Importera ett WSDL-dokument

- Bygga interoperabla applikationer genom att anpassa till Web Services Interoperability (WSI)-standarder
- Införliva Web service proxies och adapters
- Implementera en Web service end point med JAX-WS

Anpassa JAX-WS Web services

- Driftsätta en Web service WAR-fil
- Spärra trafik mellan Web services och klienter

Exponera Plain Old Java Objects

(POJO:s) som Web Services

Implementera code-first Web services

- Välja mellan WSDL-first och code-first Web services
- Generera portabla artefakter med JAX-WS
- Säkra underhållbarhet med proxies och adapters

Förbättra genererad WSDL

- Kommentera Java-tjänster
- Driftsätta end points

Implementera Web Service-klienter i Java

Generera klientkod från WSDL

- Komma åt Web services genom deras WSDL
- Skapa klient-källfiler från WSDL
- Anpassa genererade källfiler med JAX-WS

Synkrona, polling- och asynkrona tjänster

- Designa och skapa envägstjänster och -klienter
- Skriva multitrådade klienter
- Avbryta och modifiera SOAP-meddelanden

Erbjuda AJAX på serversidan med RESTful Web Services för interaktivitet

Tillståndslös bearbetning av XML-frågor

- Bygga RESTful Web services med JAX-WS
- Implementera en Provider

Lätta klienter

- Anropa Web services med Dispatch API
- Bearbeta mottagna XML-meddelanden

Stödja AJAX-klienter

- Asynkront Javascript och XML
- Erbjud interaktivitet på klientsidan

Driftsäkra Web Services

- Signera Web service-förfrågningar för att de inte ska tillbakavisas
- Säkra Web service-meddelandens konfidentialitet
- Autentisera och auktorisera klienter
- Ge autentiseringsinformation till Web services