

Program szkolenia:

Web Services w Java

Informacje:

Nazwa:	Web Services w Java
Kod:	Java-ws
Kategoria:	Java i JVM
Czas trwania:	5 dni
Forma:	50% wykłady / 50% warsztaty

Szkolenie kierowane jest do programistów Java warstwy biznesowej i integracji, którzy chcą nauczyć się tworzyć Web Service w standardach JEE. Tak w przypadku systemów homogenicznych jak i heterogenicznych.

Celem szkolenia jest nabycie umiejętności wytwarzania Web Service w Java. Tak na wysokim poziomie jak i za pomocą API niskopoziomowych - często niezbędnych w przypadku integracji z gotowymi usługami Web Service wytworzonymi poza standardami Java.

Na szkoleniu poznajemy zarówno sposoby automatycznego określania kontraktów WSDL, jak i dokładnie ćwiczymy tworzenie WSDL od podstaw w celu umożliwienia pracy zespołów systemów heterogenicznych (podejście contract first). Także ze względu na potencjalne problemy integracyjne przedstawiane są dokładnie standardy tak samych Web Service (SOAP, WSDL), jak i standardy wyższego rzędu regulujące użycie technologii pobocznych WS-* (WSIT).

Uczestnicy uczą się także posługiwania tzw. RESTful Web Services w oparciu o JAX-RS (od JEE6) aby łatwo wystawiać usługi, czy integrować się z AJAX.

Szczegółowy program:

1. Wprowadzenie do Web Service

- 1.1. Czym jest Web Service?
- 1.2. Architektura SEO
- 1.3. Zalety i wady Web Service
- 1.4. Podstawowe standardy Web Service
- 1.5. Standardy Java dotyczące Web Service

2. Szybkie tworzenie Web Service

- 2.1. JAX-WS jako następca JAX-RPC
- 2.2. Podstawy tworzenia usług w JAX-WS
- 2.3. Deployment usług JAX-WS
- 2.4. Automatycznie wygenerowany WSDL
- 2.5. Podgląd komunikatów WS (soapUI)
- 2.6. Klient JAX-WS

3. Kontrola mapowania klas na XML

- 3.1. Czym jest Marshalling i unmarshalling (JAX-B)
- 3.2. JAX-B a JAX-WS
- 3.3. Wprowadzenie do XML
- 3.4. Kontrola mapowania przez adnotacje
- 3.5. API transformacji w JAX-B
- 3.6. Standard XML Schema
- 3.7. Kontrola mapowania przez XML Schema

4. Samodzielne tworzenie komunikatów Web Service

- 4.1. Po co tworzyć komunikaty samodzielnie?

4.2. Podstawy API XML (JAX-P)

4.3. Format wiadomości SOAP

4.4. SAAJ - wsparcie tworzenia SOAP

5. Niskopoziomowa komunikacja z JAX-WS

5.1. Po co komunikacja niskopoziomowa?

5.2. Niskopoziomowy serwer JAX-WS

5.3. Niskopoziomowy klient JAX-WS

5.4. Komunikacja niskopoziomowa a WSDL

6. Definiowanie kontraktów WSDL

6.1. Web Service Description Language - po co?

6.2. Podejście contract first

6.3. Czemu WSDL 1.1 a nie 2.0?

6.4. Struktura WSDL 1.1

6.5. Contract first a JAX-WS - tworzenie usługi

6.6. Adnotacje JAX-WS sterowania treścią WSDL

6.7. Zewnętrzne sterowanie mapowaniem - bindings

7. Pozostałe elementy JAX-WS

7.1. Wstrzykiwanie zależności

7.2. Metody cyklu życia komponentu JAX-WS

7.3. Obsługa wyjątków

7.4. Handlerzy

7.5. Podsumowanie modeli komunikacji JAX-WS

8. Zaawansowane aspekty JAX-WS

8.1. Stanowe WS

8.2. Szyny danych - ESB

8.3. Przegląd technologii WS-*

8.4. WS-* a serwery WS

8.5. Standard WSIT

8.6. WS-* a WSDL

8.7. Features - wsparcie JAX-WS dla WS-*

8.8. Bezpieczeństwo WS

8.9. Transakcje długoterminowe

8.10. Przegląd biblioteki Camel

9. Przegląd rejestrów usług Web Service

9.1. UDDI

9.2. eb-XML

9.3. Ujednolicone API - JAX-R

10. RESTful Web Services

10.1. Idea RESTful WS

10.2. Protokół HTTP

10.3. Zastosowanie REST - mapowanie metod HTTP na logikę

10.4. Standard JAX-RS (od JEE6)