

Kod szkolenia: J/WSX

Tytuł szkolenia: Przetwarzanie XML i tworzenie usług Webservice na platformie Java

Adresaci szkolenia:

Szkolenie adresowane jest do programistów Java, pragnących tworzyć usługi webservice oraz poznać technologie przetwarzania XML, które są podwalinami usług webservice.

Cel szkolenia:

Celem szkolenia jest zapoznanie jego uczestników z arkanami tworzenia usług webservice wraz z technikami przetwarzania XML które są podwalinami usług WS. Uczestnicy szkolenia poznają dogłębnie język XML, definicję struktury dokumentu za pomocą DTD i XSD. Nauczą się przetwarzania XML za pomocą SAX, StaX, DOM, XSLT, oraz korzystania z odpowiedniego podejścia, w zależności od zastosowania. Dowiedzą się jak mapować obiekty Java do plików XML przy pomocy specyfikacji JAXB. Poznają również gamę dostępnych bibliotek alternatywnych.

W zakresie usług WS, dowiedzą się jak programować z wykorzystaniem niskopoziomowego api SAAJ, poznając dogłębnie SOAP i WSDL. Nauczą się tworzyć usługi przy pomocy wysokopoziomowego API JAX-RPC i najnowszego JAX-WS. Zaznajomią się z koncepcją rejestrów usług i stowarzyszoną specyfikacją JAX-R. Poznają zaawansowane zagadnienia, związane z bezpieczeństwem i współpracą między-platformową, a także zagadnienia organizacji usług, takie jak SOA, BPEL.

Wymagania:

Od uczestników wymagana umiejętność programowania w języku Java (do poznania na kursie J/JP), zalecana znajomość platformy Java Enterprise Edition.

Mocne strony szkolenia:

Program obejmuje bardzo szczegółowo zagadnienia związane z usługami Webservice. Oprócz zagadnień programistycznych, przedstawia tematykę w szerokim ujęciu infrastruktury IT w firmie. Wiedza zawarta w programie ma duży walor praktyczny.

Parametry szkolenia:

5*7 godzin wykładów i warsztatów w proporcji 1/3. W trakcie warsztatów oprócz prostych ćwiczeń projektowana i implementowana jest prosta kompleksowa aplikacja.

Wielkość grupy: maks. 8-10 osób.

Polecane szkolenia poprzedzające:

J/JP

Program szkolenia:

1. XML

- a. konstrukcja języka XML
- b. definiowanie struktury dokumentu przy pomocy DTD
- c. definiowanie struktury dokumentu przy pomocy XSD

2. Java API for Xml processing (JAXP) i inne API związane z przetwarzaniem XML)

- a. Rodzaje przetwarzania XML: strumieniowe (pull i push), oparte na drzewie, oparte na wzorcach.
- b. Wybór właściwego sposobu przetwarzania XML
- c. Omówienie specyfikacji wchodzących w skład JAXP
- d. Przetwarzanie zdarzeniowe na podstawie SAX
 - i. Wprowadzenie do SAX i dostępne implementacje
 - ii. Koncepcja i konfiguracja parsera

- iii. Zdarzenia dotyczące zawartości (ContentHandler)
- iv. Zdarzenia dotyczące błędów (ErrorHandler)
- v. Zdarzenia dotyczące DTD (DTDHandler)
- vi. Zdarzenia rezolwencji encji (EntityResolver)
- vii. Lokalizacja źródeł zdarzenia (DocumentLocator)
- e. Przetwarzanie drzew na podstawie DOM, oraz JDOM
 - i. Wprowadzenie do DOM
 - ii. Poziomy DOM
 - iii. Kluczowe klasy i interfejsy
 - iv. Wczytanie dokumentu
 - v. Nawigacja po dokumencie
 - vi. Konstruowanie dokumentu
 - vii. Walidacja dokumentu
 - viii. DOM , JDOM, dom4J
- f. Biblioteka StAX
 - i. API Stax
 - ii. Cursor
 - iii. Iterator
 - iv. Rodzaje i obsługa zdarzeń
 - v. Filtry
 - vi. Czytanie i pisanie strumieni XML
 - vi. Korzystanie ze Stax(XMLInputFactory, XMLOutputFactory, XMLEventFactory)
- g. Przetwarzanie wzorców : XSL(XPath, XSLT, XSL-FO)
 - i. Wprowadzenie do XSL
 - ii. Omówienie XPATH

- iii. Wyrażenia
 - iv. Template
 - v. Adresowanie
 - vi. Wyrażenia złożone
 - vii. Transformacja dokumentu przy pomocy XSLT
 - viii. Formatowanie dokumentów za pomocą XSL-FO
 - ix. XQuery
3. Java Api for XML Binding(JAXB)
- a. Architektura JAXB
 - b. Mapowanie Klasa Java XML Schema
 - c. Marshalling, Unmarshalling
 - d. Walidacja
 - e. Modyfikacja standardowych mapowań za pomocą adnotacji
4. Inne biblioteki związane z XML: Apache XmlBeans, JiBX, Castor, Jelly
5. Idea i zasada działania usług webservice
- a. SOAP
 - b. XML-RPC jako alternatywa SOAP
 - c. WSDL
 - d. UDDI
 - e. ebXML
6. Przegląd oprogramowania typu „webservice Stack”: (Axis ,Celtix ,Glue,JBossWS,XFire ,JAXWS)
7. SAAJ
- a. Wprowadzenie do SAAJ
 - b. Wiadomość SAAJ

- c. Budowa
 - d. Tworzenie wiadomości
 - e. Manipulacja zawartością wiadomości
 - f. Dodawanie contentu
 - g. Dodawanie nagłówków
 - h. dodawanie dokumentu
 - i. Przestrzenie nazw
 - j. Ustanawianie połączenia, wysyłanie załączników odczyt wiadomości
 - k. Dodawanie załączników (SOAP Attachments vs MTOM)
 - l. Przesyłanie informacji o błędach (SOAPFaults)
8. JAX-WS, JAX-RPC
- a. Idea JAX-RPC i JAX-WS
 - b. Wspierane typy
 - c. różnice między JAX-RPC a JAX-WS
 - d. Tworzenie usługi w JAX-WS
 - i. Na podstawie SEI (Webservice)
 - ii. Opartej na przetwarzaniu SOAP(WebserviceProvider)
 - e. Rodzaje wiązań
 - f. Klasa usługi a WSDL
 - i. Tworzenie usługi WS na podstawie klasy
 - ii. Tworzenie usługi WS na podstawie WSDL
 - g. Kompilacja i generowanie artefaktów
 - h. Konfiguracja usługi za pomocą adnotacji (JSR-181)
 - i. Tworzenie Webserviców typu REST w JAX-WS
 - j. Przechwytywanie komunikatów (Handlers)

- k. Stworzenie klienta do usługi WS
 - i. Dynamic Proxy Client
 - ii. Dynamic Invocation Interface Client
 - iii. Klient oparty na Stub'ie
 - iv. Klient JAX-WS
 - l. Features
 - m. EJB jako WS
 - n. WS na platformie Axis
 - o. Stworzenie klienta na platformie innej niż Java
9. JAX-R
- a. Pojęcie rejestru
 - b. Architektura JAX-R
 - c. Łączenie się z rejestrem
 - d. Tworzenie zapytań do rejestru
 - e. Publikacja w rejestrze
 - f. Taksonomie
 - g. Korzystanie za jax-r w środowisku JEE
10. Zagadnienia architektury i organizacji usług
- a. Webservice orchestration – wstęp do BPEL
 - b. REST
 - c. SOA
11. Zagadnienia interoperability WS-I, WSIT
12. Przegląd innych specyfikacji Java stowarzyszonych z tematem WS :
JBI, JSR 109
13. Zagadnienia bezpieczeństwa WSS, XWS-security

14. Poglądowe omówienie specyfikacji WS: WS-Addressing, WS-Atomic Transaction, WS-Business Activity, WS-Coordination, WS-Eventing, WS-Metadata Exchange, WS-Notification, WS-ReliableMessaging, WS-Policy, WS-Secure Conversation, WS-Security Policy, WS-Security, WS-Trust ,WS-Transfer

