

# Advanced Java

---

GWE10807

5 day

## Table of Contents

[Prerequisites](#)

[Course Summary](#)

## Prerequisites

Attendees should have practical experience developing Java applications.

## Course Summary and Lessons

The five-day course begins with a discussion of web application architecture. A major part of the course is spent on Servlets, JavaServer Pages (JSP), and JavaServer Faces (JSF). It then covers interacting databases, initially looking at JDBC, but then moving to JPA and how its implementations can handle mapping between objects and the relational model. Throughout the course, students will create code for an online store. Students will learn not only specific topics and APIs but also how to fit the pieces together into a complete application.

### Session JEE Application Architecture

#### Lesson Technical Overview of JEE

- What is JEE?
- Common Themes In JEE Framework
- JEE Containers and Components
- Servlets
- Java Server Pages (JSP)
- JavaServer Faces (JSF)
- EJBs and Web Services
- JEE Containers
- JEE Application Modules
- The JEE 5 Specification
- JEE Platform Roles

#### Lesson JEE Application Architectures

- JEE and the Web
- JEE 1st Generation DB-centric Web Application
- Adding JSPs to Separate Presentation
- Eliminating Java code from view
- The Model 2 Architecture
- Using EJBs
- JEE Infrastructure Supporting Web Applications Lesson Review

### Session Web Applications

#### Lesson Understanding Web Applications

- JEE Application Modules
- The Truth about Archives
- Enterprise Application Archive (EAR)
- Enterprise JavaBean Archive (JAR)

- JEE Application Client (JAR)
- Resource Adapter Archive (RAR)
- Web Application Archive (WAR)
- Directory Structure

#### Lesson Configuring Web Applications

- Mapping an HTTP Request to a Resource
- The web.xml File
- Structure
- Declaring Servlets and JSPs
- Servlet Mapping
- Servlet Init Parameters
- Web Application init Parameters
- Welcome Page
- Error Page

### Session Developing Servlets

#### Lesson Introduction to Servlets

- Servlet Overview
- Life Cycle of Servlets
- Servlet Lifecycle is Handled by Web Container
- HttpServlet
- Writing the init Method
- HttpServlet doXXX Methods
- Writing a Simple HTTP Servlet
- HttpServletRequest Methods
- ServletResponse
- HttpServletResponse
- Servlet I/O Classes
- Return a Status Code

- Building the Output Document
- Sending Binary Content

#### **Lesson Processing Input Data**

- Form Processing with Servlets
- HTML Form
- LoginServlet doPost

#### **Lesson Server-Side Control**

- Request Dispatcher
- Forward the processing
- Passing Processing on and Getting it Back
- Servlet Runs Within Web Container Environment
- Several Options for Receiving Data
- Init Parameters and Attributes
- ServletConfig; ServletContext
- Servlet Variables are Scoped
- HTTP Request Information
- Several Options for Sharing Data
- configuration and Context
- Servlet Variables
- HttpServlet Request
- Threading and Data
- Threading and Data 2

#### **Lesson Client Side Control**

- Output Buffering
- Setting Status Codes; Setting Headers
- sendRedirect
- Disabling Client Caching
- Supporting Persistent Connections
- Setting Content Length
- Dynamic Content Pushing

#### **Lesson Maintaining Client State – Sessions**

- Session Management
- Tracking Problem – Stateless HTTP
- Data Problem – Session Data
- Solving the Tracking Problem
- Cookies
- Cookie Behavior; Retrieving Cookies
- servlet to set Cookies
- Servlet to Show Cookies
- URL Rewriting
- Solving the Data Problem
- Web Container Manages Session Instances
- Sessions with Cookies
- Cookie-Based Sessions
- Basic Session Implementation
- Cookie Detection is not Standardized
- Getting Rid of Http Sessions
- Session with URL Rewriting

#### **Lesson Application and Session Events**

- Event Listener Model
- Life Cycle Events in a Web-Application
- Declare the Listener
- Type of Events
- Context Listeners; Session Listeners
- Session Listeners for Session-Objects

### **Session Filters**

#### **Lesson Overview of Filters**

- What is a Filter
- Single Filter
- Filter Objects
- doFilter Method
- init Method
- Filter Life Cycle
- Cascading Filters

#### **Lesson Filtering Requests and Responses**

- Request Wrapper
- Process the Request
- Examples of Request Filters
- Filter the Response
- Response Wrapper

### **Session Developing JavaServer Pages**

#### **Lesson Introduction to JavaServer Pages**

- Separating Presentation from Model
- Java Server Page (JSP): An Extension of Servlet
- Lifecycle of a JSP
- Example JSP
- JSP Syntax Consists of Three Types
- JSP Scripting: Declarations; Expressions; Scriptlets & Directives
- The session Attribute
- The errorPage/isErrorPage Attribute
- JSP Actions
- JSP Actions: Include/Forward
- Typical JSP Access Model
- JSP Action: useBean
- Implicit Objects
- JSPs or Servlets?

#### **Lesson JSP Implicit Objects**

- Implicit Objects
- Page Object; Config Object
- Request Object; Response Object
- Out Object
- Output Buffer
- Session Object
- Application Object
- PageContext Object
- Attributes
- Session Attributes
- Exception Handling

#### **Lesson Actions, Java Beans,™ and Custom Tags**

- Standard Actions
- Forwarding; Including
- Using JavaBeans™ and JSP
- Declaring to use a Bean
- Using a Bean, Example
- Setting and Getting Properties
- What are Custom Tags?
- Create and Use a Custom TagLib

## Lesson JSPs in Depth

- JSP 2.1 Specification
- The web.xml in JEE 5
- <jsp-config>
- JSP Format Rules
- JSP Error Pages

## Lesson The Unified Expression Language

- The Unified Expression Language
- Enable/Disable EL
- Variables
- Literals and Operators in Unified EL
- Implicit Objects in Unified EL
- Reserved Words
- Unified EL Functions
- Developing the Function
- Declaring the Function in the TLD
- Using the Function
- Pre-Built Unified EL Functions

## Session JSTL

### Lesson JSTL Introduction and Core Library

- Introduction to JSTL
- Expression Language (EL) in JSTL
- Review of JSP Bean Tags
- Tag Collaboration
- JSTL Core
- <c:choose> <c:when> <c:otherwise>
- <c:if>; <c:import>; <c:forEach>; <c:forTokens>; <c:out>; <c:param>; <c:catch>; <c:redirect>; <c:remove>; <c:set>;
- Using c:set to pass HTML
- <c:url>

### Lesson JSTL Format Library

- JSTL Format
- <fmt:requestEncoding>; <fmt:setLocale>; <fmt:timeZone>
- <fmt:setTimeZone>; <fmt:bundle> <fmt:setbundle>; <fmt:message>
- <fmt:formatNumber>; <fmt:parseNumber>
- <fmt:formatDate>; <fmt:parseDate>

### Lesson JSTL SQL Library

- JSTL SQL
- <sql:query>; <sql:update>; <sql:transaction>; <sql:param> <sql:dateParam>

### Lesson JSTL XML

- JSTL XML
- <x:choose> <x:when> <x:otherwise>
- <x:out>; <x:if>; <x:forEach>; <x:parse>; <x:set>; <x:transform>; <x:param>

### Lesson Simple Tag Handlers

- Tag Files
- Simple Tag Handlers
- Tag File Location
- Tag-Files Outside of a Library
- Tag Library Descriptors
- The TLD File

- The <tag-file> Element
- Tag File with Attributes
- jsp:attribute Element
- Using Attributes
- jsp:doBody
- jsp:invoke

## Session Additional JEE Components

### Lesson Working With Enterprise JavaBean™ 3.0

- Defining Enterprise JavaBeans
- JavaBeans™ vs EJBs
- EJB Architecture Overview
- EJB Container & Types of EJBs
- Enterprise Bean; Session Beans
- Entities
- Message-Driven Beans
- The Decorator
- Deployment Descriptor
- EJB-Jar File

### Lesson Web Services in JEE – WSEE

- What are Web Services?
- Web Services Architecturally
- XML and Web Service APIs
- Web Services for JEE - WSEE
- Servlets as Web Services
- Stateless Session EJBs as Web Services
- Routing SOAP requests to an EJB
- WSDD

## Session Accessing JEE Resources

### Lesson Java Naming and Directory Interface (JNDI)

- Naming Services Overview
- Unified Heterogeneous Namespace
- JNDI
- JNDI Implementation Architecture
- Benefits
- Example of JNDI Usage

### Lesson Dependency Injection

- Dependency Injection (DI)
- DI to Handle Access
- DI: Inversion of Control
- DI: Implementation
- DI: JEE
- DI: JEE Example
- JNDI vs DI

## Session Persistence in JEE

### Lesson Java Persistence API (JPA) and Entities

- Object-Relational Mapping (ORM)
- Typical ORM Approaches
- ORMs use Lazy Loading
- ORM without Lazy Loading
- Benefits of ORM Framework Like Hibernate
- Example of Lazy Loading
- Loading an Entire Object Graph

- JPA Overview
- JPA Package
- JPA Configuration Files
- Example of persistence.xml
- Mapping Objects to the Database
- Example of Annotated Mapping
- Mapping with Annotations
- JPA Example
- EntityManager Works With Entities

#### **Lesson JDBC and Its Position in JEE**

- JDBC Versions
- The JDBC API
- JDBC in JEE
- Programming with DataSources
- JNDI names
- DataSource Programming Best Practices

#### **Lesson JDBC Data Access API**

- Structured Query Language (SQL)
- Statements & Statement
- PreparedStatement
- PreparedStatement
- ResultSet
- Executing Inserts, Updates, and Deletes
- Mapping SQL Types to Java Types
- CallableStatement

#### **Lesson The DAO Pattern**

- Data Access Object (DAO)
- DAO Structure
- DAO Example: ProductDAO
- The DAOFactory
- Complete UML Diagram
- Using the ProductDAO

#### **Session Security**

##### **Lesson JEE Security**

- Typical JEE App Server Security Services
- Java 2 Platform Security Model
- Java 2 Security in JEE
- JAAS Authentication: Who are you?
- JEE Security Overview
- Authorization: Are you allowed access?
- High-Level Trace of JEE Authorization
- Deployment Descriptors Play a Large Role
- Declaring Secure Resources
- Declaring the security roles
- Typical JEE App Server Security Services
- Security on the Web; Secure Web Traffic
- SSL In Action
- Responsibilities For Security
- CMS: Declaring HTTPS
- Authentication Challenge Mechanisms

##### **Lesson Web Application Security Overview**

- Attacks are Constant and Changing
- Open Web Application Security Project
- Assets are the Targets

- The Context for Defensive Coding
- Attackers Not Hackers
- Cross-Site Scripting (XSS): Description
- Defending Against XSS Attacks
- SQL Injection: Description; Example & Drill Down
- Defending Against SQL Injection Attacks

#### **Lesson Handling Untrusted Input**

- Unvalidated Input: Description
- Protecting a Web Resource
- Defending a Web Application
- Defending a Web Application/Resource
- Responding to Error State
- Best Practices for Untrusted Data
- Additional Types of Attacks

#### **Session Additional JEE Topics**

##### **Lesson Transactions**

- Transaction Definitions
- The ACID Transaction Properties
- Transaction Lifecycle
- Overview of a Transactional System
- JEE Transaction Support (JTA)

##### **Lesson Other JEE APIs**

- Java Message Service (JMS)
- When is Messaging Used?
- Two Messaging Models
- More On Publish/Subscribe
- Logical View of Publish/Subscribe
- More On Point-to-Point (P2P)
- Logical View of Point-To-Point
- Message Servers
- JavaMail; JavaMail Architecture
- XML
- An XML Document
- JEE and XML

#### **Appendix: Introduction to Annotations**

- Annotations Overview
- Working with Java Annotations
- Example of Using @Override
- Annotations Used Extensively in Many Technologies
- Declaring Persistence in Hibernate