

Course Description: This course is an introduction to writing JEE-compliant Web applications using JBoss 4 and the JBoss Eclipse IDE. An overview of JEE technology is provided, followed by hands-on experience with JNDI, JDBC, Java servlets, and JavaServer pages. Other topics covered include servlet filters, custom JSP tags, JavaMail, and an overview of JavaServer Faces (JSF).

Who Should Attend: Experienced Java programmers and software engineers preparing to write components for JEE Web applications hosted on JBoss 4 Application Server.

Prerequisites: Students should be comfortable with Java programming and object-oriented concepts. A minimum of six months coding experience is suggested. In addition, students should be familiar with writing simple Web pages using HTML. Prior experience using SQL and/or JDBC will be helpful.

Benefits of Attendance: Upon completion of this course, students will be able to:

- Start, stop, and configure JBoss Application Server
- Write, deploy, and test JEE components using the JBoss Eclipse IDE development tool
- Use JNDI to access JDBC data sources
- Write and deploy servlets and JavaServer pages
- Use JDBC to read and update a database
- Create and process HTML forms
- Work with cookies and HTTP sessions
- Assemble and configure a J2EE-compliant Web application
- Use servlet filters for pre- and post-processing HTTP requests
- Create custom JSP tags
- Use the JavaMail API to send email from Web applications
- Create security principals and roles
- Apply security to Web pages
- Implement the Model View Controller architecture
- Write simple applications using the JavaServer Faces framework

Course Outline:

Overview of Java EE

Java Editions
Characteristics of "Enterprise" Computing
Java EE Technologies
Multi-Tier Architectures
Advantages of Multi-Tier Architectures
Container-Based Approach
Java EE Application Models

Introduction to JBoss Eclipse IDE

What is JBoss Eclipse IDE?
Starting JBoss Eclipse IDE
Configuring an Installed JRE
Setting Compiler Preferences
Configuring the JBoss Test Environment
Verifying Server Start-Up
The JMX Console
Stopping and Starting the JBoss Server
Web Applications
Creating a Project
Setting Up a Packaging Configuration
Setting up an XDoclet Configuration
Deploying the Web Application
Testing the Web Application

Servlets

A Simple Servlet
Configuring Servlets
Servlets in JBoss Eclipse IDE
Servlet Initialization Parameters
Generating and Validating Forms
Servlets and Threads
Other Settings in `web.xml`

Session Management

Cookies
Sessions
Session Id's
Session Management
Session Management Example
Invalidating Sessions
Configuring the Session Timeout

JavaServer Pages

JavaServer Pages
A Simple JSP
JavaServer Pages in JBoss Eclipse IDE
JSP Core Tags

JSP Directives
JSP Actions
JSP Example with Forwarding
JavaServer Pages and JavaBeans
JSP with JavaBean Example
Running the JSP Bean Example
Creating a New JSP

Custom JSP Tags

Using Custom Tags
Types of Tags
Defining Tags
The `tag` Element
Building the Examples
Simple Tags
Tags with Attributes
Using JSP Expressions as Attributes
Including the Tag Body
Optionally Including the Body
Including the Body Multiple Times
Creating a New Tag Handler

Web Application Security

HTML Form for Survey Application
HTML Code for Survey Form
Servlet Code for the Survey Application
JavaBean Class for the Survey Application
Running the Survey Application
Java EE Security
Configuring Authentication for Web Applications
Configuring Authorization for Web Applications
Securing the Survey Application

Java Naming and Directory Interface

What is JNDI?
Benefits of JNDI
Naming Services
Directory Services
Using JNDI
Context Operations
JNDI Utility Class
JNDI Example
Running the JNDI Example
Naming Exceptions
Creating a New Client Program

Database Access using JDBC

A Simple JDBC Program
JDBC Driver Types
Connection Pools
JDBC Data Sources
Creating Tables
Running the JDBC Examples
Executing a Query
Deleting a Row

Design Concepts for Web Applications

Architecture and Design
Tiered Architectures
Model-View-Controller Architecture
J2EE Design Patterns
Composite View Pattern
Composite View Strategies
Running the Demo Application
View Helper Pattern
Front Controller Pattern
Intercepting Filter Pattern

Servlet Filters

What is a Filter
Sample Filter
The Filter API
Initializing Filters
Blocking the Response
Modifying the Response
Running the Filter Examples
Creating a New Filter

JavaMail

JavaMail
Example - Send Mail
Example - Read Mail
Example - Read Mail
Running the Examples

JavaServer Faces

What is JavaServer Faces?
JSF Development Roles
Developing a JSF Application
JSF Components
Component Tags
Validators
Example - Creating a Form
Optional Settings in `web.xml`

Backing Beans
Example - Processing a Form
Backing Bean Class
The `faces-config.xml` file

Appendix A: Web Resources

Java Technology
JBoss

Appendix B: HTML Reference

Introduction
A Simple HTML Document
Basic Tags
Formatting Tags
Links
Forms

Appendix C: Web Accessibility

What is Accessibility and Why It is Important?
What is Section 508?
Accessibility Initiatives and Related Legislation
Types of Disabilities
Assistive Technologies
Benefits of Accessible Design
General Coding Practices
Non-Text Elements
Multimedia Presentations
Color and Style Sheets
Image Maps
Tables
Frames, Frequency and Equivalents
Scripting
Programming
Forms
Repetitive Navigation Links
Timed Responses
Other Recommendations

Appendix D: A JSP Template Mechanism

A Sample Application
A JSP Template Mechanism
Implementing the Template Mechanism with Custom JSP Tags
Classes in the Sample Application
Tag Library Descriptor
Running the Sample Application