

**Course Description:** This course is an introduction to writing JEE-compliant Web applications using Oracle WebLogic Server 10.x and Oracle Workshop for WebLogic. 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 Oracle WebLogic 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 WebLogic Server
- Write, deploy, and test Java EE components using the Oracle Workshop development tool
- Use JNDI to access JDBC data sources
- Write and deploy servlets and JavaServer pages on WebLogic server
- 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  
Java EE Web Application Model

### Introduction to Oracle Workshop

What is Oracle Workshop?  
Starting Oracle Workshop  
Configuring the WebLogic Test Environment  
Starting and Stopping WebLogic Administration Console  
Setting Preferences in Oracle Workshop  
Creating a Dynamic Web Project

### Servlets

A Simple Servlet  
Web Applications  
Configuring Servlets  
Running Servlets in Oracle Workshop  
Servlet Initialization Parameters  
Generating and Validating Forms  
Servlets and Threads  
Other Settings in web.xml  
Creating a New Servlet

### 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  
Running JavaServer Pages in

Oracle Workshop  
JSP Syntax  
Configuring JavaServer Pages  
JSP Directives  
JSP Actions  
JSP Example with Forwarding  
JavaServer Pages and JavaBeans  
JSP With JavaBean Example  
JavaBean Class  
Running the JSP Bean Example  
Creating a New JSP

### Custom JSP Tags

Using Custom Tags  
Types of Tags  
Defining Tags  
The tag Element  
Simple Tags  
Simple Tags - Example  
Tags with Attributes  
Tags with Attributes - Example  
Using JSP Expressions as Attributes  
Including the Tag Body  
Including the Tag Body - Example  
Optionally Including the Body  
Including the Body Multiple Times  
Including the Body Multiple Times - Example  
Running the Examples

### 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  
Users and Groups  
Adding Users in the Administration Console  
Authentication  
Configuring Authentication for Web Applications  
Authorizing Access to Resources in a Web Application

Web Application Security – Example

### 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 Standalone Program

### Database Access using JDBC

A Simple JDBC Program  
JDBC Driver Types  
Connection Pools  
JDBC Data Sources  
Data Source Example  
Configuring JDBC Data Sources  
Running the JDBC Examples  
Using JDBC in a Servlet  
Using JDBC in a JSP

### Design Concepts for Web Applications

Architecture and Design  
Tiered Architectures  
Model-View-Controller Architecture  
Java EE Design Patterns  
Composite View Pattern  
Composite View Strategies  
Running the Demo Application  
Composite View Pattern – Implementation  
View Helper Pattern  
View Helper Pattern – Implementation  
Front Controller Pattern  
Front Controller Pattern – Implementation  
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  
Running the Examples

### JavaServer Faces

What is JavaServer Faces?  
JSF Development Roles  
Developing a JSF Application  
Validators  
Example – Creating a Form  
Running the First Example  
Backing Beans  
Example – Processing a Form  
Backing Bean Class  
The faces-config.xml file  
Running the Second Example

### Appendix A: Web Resources

Java Technology  
WebLogic

### 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

### Appendix E: Web Services

Service-Oriented Architecture  
Web Service Components  
Simple Object Access Protocol  
SOAP Message Format  
The SOAP Envelope  
WSDL and UDDI  
Example  
Creating a Web Services Project  
Running the Example  
Testing with the WebLogic Test Client