



Course Overview

Today java is one of the most mature and commonly used programming languages for building enterprise software. Java is providing enterprise solutions to small, medium and large enterprise throughout the world and is a leading player in mobile applications. The need of java professionals is growing day by day, which is in itself, a testimony to its success.

This course elaborates on the use of the Java Platform, Enterprise Edition (Java EE) technology design patterns in designing web services and discusses various web services features, such as Exception Handling and Security provided by the Java Platform, Enterprise Edition.

The focus of the course is on building web-enabled applications that employ Servlets, Java Server Pages and Tag Libraries to exploit the services provided by the Java EE platform.

This course covers architectural design issues as well as specific coding models for Java EE components, and is up to date with the latest Java EE 5, JSP 2.1 and Servlet 2.5 specifications.

Since coding and deployment files are standardized by the Java EE specifications, students may readily apply the skills learned in this class to write code for any compliant server like Apache Tomcat, JBoss, WebSphere, Oracle, WebLogic and many others.

Audience Profile

This course is intended for programmers who are evaluating J2EE framework, and provides various resources where they can obtain specific skills for creating applications, components, and services built using Java skills & programming.

At Course Completion

After completing this course, students will have a fundamental understanding of how to:

- Make effective use of Java generic types
- Write multi-threaded Java applications
- Use the Reflection API for highly generic tasks, discovery, or code-generation
- Use standard annotations and develop custom annotations to express meta-data in Java source files
- Communicate between processes using network sockets
- Document and package a Java application.
- Capture configuration and debugging information using the Java Logging APIs.
- Use many of the new enhancements added to the Java API.
- Use features of the new I/O API.
- Choose appropriate data structures from the Java Collection API.
- Sort and search arrays and lists using a variety of techniques.
- Serialize Java objects.
- Write TCP/IP Client Server applications using sockets.
- Execute methods on a remote object using RMI.
- Perform database queries and updates using JDBC

Advance Java and J2EE (Java 2 Enterprise Edition)

Environment: JAVA 1.8

Course Duration: 40 Hours

- **Introduction of J2EE**
 - Introduction to Enterprise Edition
 - Distributed Multitier Applications
 - J2EE Containers
 - Web Services Support
 - Packaging Applications
 - J2EE 1.4 APIs
- **Server Programming using Servlets**
 - What is Web Application?
 - What Is a Servlet?
 - Servlet Life Cycle
 - Sharing Information
 - Using Scope Objects
 - Controlling Concurrent Access
 - Getting Information from Requests
 - Constructing Responses
 - Filtering Requests and Responses
 - Programming Filters
 - Customized Requests and Responses
 - Specifying Filter Mappings
 - Invoking Other Web Resources
 - Including Other Resources in the Response
 - Transferring Control
 - Accessing the Web Context
 - Maintaining Client State
 - Session Management
 - Session Tracking
- **Java Server Pages (JSP)**
 - Java Server Pages Technology
 - The Life Cycle of a JSP Page
 - Translation and Compilation
 - Creating Static Content
 - Response and Page Encoding
 - JSP Basic Tags and its uses
 - Creating Dynamic Content
 - Using Objects within JSP Pages

- Expression Language
- Deactivating Expression Evaluation
- Implicit Objects
- Custom Tags
- JavaBeans Components
- Reusing Content in JSP Pages
- **Java Mail API**
- **Annotations**
- **XML and Web Services**
- **Hibernate 3.x**
 - Introduction to ORM and Hibernate
 - Hibernate Architecture
 - Hibernate Configuration
 - Introduction to Session and SessionFactory
 - Persisting Objects using Hibernate
 - Hibernate Query Language
 - Inheritance Mapping
 - Bidirectional Association Mapping
 - Caching Introduction
 - Implementing Second Level Caching using EHCACHE
 - Integrating Spring & Hibernate, Struts & Hibernate, JPA & Hibernate
- **Spring 3.x**
 - Introduction to Spring
 - Understanding Inversion of Control (IOC) and
 - Dependency Injection (DI)
 - Spring Architecture
 - Implementing IOC in Spring
 - Understanding Aspect Oriented Programming (AOP)
 - Implementing AOP in Spring
 - Introduction to Template Design Pattern
 - Simplifying Data access using JDBC Template
 - Abstracting Data Access Layer using DAO
 - Introduction to Spring MVC
- **SQL Basic**
 - Introduction
 - Select, Where, Insert, Update and Delete Commands
- **SQL functions**
- **Working with dates, procedure, functions, triggers and sequences**