

## **J2EE Programming using IBM Rational Application Developer (RAD) v6**

This 5 day training course teaches students how to develop Servlet, JSP and EJB for the WebSphere Application Server v6 platform. About 50% of the time is spent on hands on labs that use the Rational Developer Studio (RAD) v6 development tool.

### **Topics**

- The core J2EE 1.4 specifications - Servlet, JSP and EJB
- Developing a business logic layer using EJB
- Developing a presentation layer using Servlet and JSP
- Mastering the Rational Application Developer (RAD) tool

### **Learning Objectives**

After completing this course, the student should be able to:

- Develop web based applications using Servlet and JSP
- Develop business logic layer using EJB
- Be productive developers in the Rational Application Developer (RAD) environment
- Deploy application code inside WebSphere Application Server v6

### **Audience**

This course is designed for Java programmers and web application designers who have interest in learning how to build or maintain enterprise applications based on the J2EE architecture using IBM's WebSphere tools.

### **Duration**

Five days.

### **Price**

\$2,495.00

## Course Outline

### 1. Introduction to Rational Application Developer

- Eclipse and its plugin architecture
- Workspace, Workbench, perspective and view
- Developing Java applications
- Compiling Java applications
- Refactoring Java code
- Searching

### 2. Introduction to J2EE and WebSphere v6

- Core J2EE concepts
- WebSphere Application Server product family
- Key capabilities of WebSphere Application Server v6

### 3. Rational Application Developer (RAD)

- J2EE development features
- Creating an Enterprise, Web and EJB project.
- Setting up a test server (WebSphere Test Environment)

### 4. Servlet Basics

- Servlet overview
- HTTP Servlet API
- Servlet life cycle
- Example
- Developing servlets in RAD

### 5. JSP Basics

- JSP overview
- JSP directives
- JSP expressions
- JSP scriptlets
- JSP declarations
- Examples
- Developing and testing JSP

### 6. Servlet Interaction Interface

- The request object
- The response object
- Cookie
- HTTP Session management

### 7. Advanced Servlet Programming

- The Model View Controller (MVC) architecture
- JDBC programming
- DataSource and connection pool
- Invoking a JSP from a Servlet
- Passing data from a Servlet to JSP

### 8. JSP EL and JSTL

- JSP Expression Language
- JSTL tags - basic tags, condition tags, iteration tags, internationalization tags.

## 9. Enterprise JavaBeans Programming - Overview

- EJB fundamentals
- Overview of EJB container and EJB server responsibilities
- Overview of entity EJBs and session EJBs
- EJB deployment descriptor

## 10. Enterprise JavaBeans Programming - Session Bean

- Session bean contexts and lifecycle
- Writing stateless session beans
- Writing stateful beans
- Examples
- Creating session beans

## 11. Enterprise JavaBeans Programming - Entity Bean

- Entity bean contexts and lifecycle
- Writing container-managed entity beans
- Writing bean-managed entity beans
- EJB QL to develop finder methods
- Container managed relationship (CMR)
- Examples
- Creating entity beans
- Mapping CMP attributes to database tables

## 12. Best Practices

- Stateful Session Beans
- Transaction Isolation Levels
- Which EJB objects to cache
- CMP EJB Pooling
- Access Beans
- Using HTTP Session
- Connection Pooling
- Java Best Practices
- WebSphere Best Practices

## 13. Deploying Enterprise Application Projects to WebSphere v6

- Exporting the Enterprise Application
- Deployment Descriptors
- Application Assemble Entries
- Installing an Application into WAS v6
- Managing WAS applications

## 14. J2EE Design Patterns

- EJB Layer Architectural Patterns
- Inter-tier Data Transfer Patterns
- Transaction and Persistence Patterns
- Client-side EJB Interaction Patterns
- Primary Key Generation Strategies

## 15. Using RAD in a Large Project

- RAD performance tuning
- Breaking up large projects
- Remote debugging

## Appendix A - Message-Driven Beans

**Appendix B - Custom JSP Tag Libraries**

**Appendix C - Java Messaging Service**