# Java EE Web Application Development using Ganymede Eclipse

# Course Summary

**Description**

This workshop will teach students to build database enabled J2EE Web programming knowledge and skills in the Java EE Ganymede Eclipse environment using Servlets, Filters, JSP and related technology. The students will learn how to build a web application using the Model-View-Controller (or MVC or Model II) design paradigm. It will also include an overview of the Apache Struts framework. Servlets, JSP and filters will be covered extensively, including programming and configuring these components. All aspects of JSP will be covered. Students will write simple JSPs, write and use JavaBeans. Various aspects of accessing data and managing state efficiently are covered, including JDBC and HTTP session management.

**Topics**

- Getting Started with JEE Eclipse and Tomcat 6.0
- Servlets
- Overview of HTML
- JavaServer Pages
- JavaBeans
- Integrating Servlet and JSP
- JDBC
- Configuring Servlet/JSP Using web.xml
- Model View Controller
- Struts Framework
- Servlet and JSP Filters
- Using Annotations in Servlets
- J2EE Architecture Overview (optional)

**Audience**

This course is intended for Java developers who want to understand, design and build Web applications using the latest Java technologies.

**Prerequisites**

The student should have a working knowledge of Java programming and some experience with HTML.

**Duration**

Five days

ProTech
protechtraining.com

*"Charting the Course ...*

*... to Your Success!"*

# Java EE Web Application Development using Ganymede Eclipse

## Course Outline

**I. Getting Started with Java EE Eclipse and Tomcat Server**
- A. Software Requirements
- B. JRE 1.5
- C. Eclipse Platform
- D. Eclipse Architecture
- E. Apache Tomcat Server 6.0
- F. JEE Web and Tomcat

**II. Servlets**
- A. What is a servlet?
- B. Typical Uses of Servlets
- C. How Servlets Work
- D. Java Servlet Architecture
- E. Servlet's Lifecycle
- F. The service() method
- G. Writing HelloServlet
- H. Deployment Descriptor
- I. The ServletRequest Object
- J. Handling Form Data
- K. doGet() vs doPost()
- L. Using doGet()
- M. Using doPost()
- N. The ServletResponse Object
- O. Session Management

**III. Overview of HTML**
- A. HTML
- B. HTML Tags
- C. HTML Document
- D. HTML Data Structure
- E. HTML Form Creation
- F. HTML <form> tags
- G. HTML <input> tags
- H. An HTML Input Form - an Example

**IV. JavaServer Pages**
- A. JavaServer Pages
- B. Why Use JSP?
- C. How JSP Works
- D. JSPs Life
- E. JSP Tags
- F. Directives
- G. Action
- H. Scripting elements
- I. Comment
- J. JSP Directive
- K. The page Directive
- L. The include Directive
- M. The taglib Directive
- N. Scripting Elements
- O. Scriptlets
- P. Expression
- Q. Declaration
- R. Comments
- S. Using JSPs Scripting Elements
- T. JSP Implicit Objects

**V. JavaBeans**
- A. JavaBeans
- B. JavaBean Structure
- C. A Simple Example
- D. Standard action
- E. <jsp:useBean>
- F. <jsp:getProperty>
- G. <jsp:setProperty>
- H. JavaBeans in JSPs

**VI. Integrating Servlets and JSPs**
- A. Calling a Servlet from JSP
- B. Using the FORM tag
- C. <jsp:include> Action
- D. <jsp:forward> Action
- E. Call a JSP from a Servlet
- F. Using RequestDispatcher
- **G.** Using the sendRedirect() Method
- **H.** sendRedirect() vs forward()

**VII. JDBC**
- A. What is JDBC
- B. Software Requirements
- C. JDBC Driver Types
- D. JDBC 1.0
- E. Load the Driver
- F. Establish a Connection
- G. Execute SQL statements
- H. Statement
- I. execute()
- J. executeQuery()
- K. executeUpdate()

# Java EE Web Application Development using Ganymede Eclipse

## Course Outline (cont'd)

L. ResultSet
M. Inserting a record
N. Updating a record
O. Deleting a record
P. Prepared statement
Q. Callable statement
R. DataSource
S. Using a DataSource

**VIII. Deployment Descriptor – web.xml**
A. Deployment descriptor
B. Order of Elements
C. Adding a Aervlet
D. <servlet> element for Servlet
   1. <servlet> element for JSP
   2. <servlet-mapping> element
E. Custom URL for a Servlet
F. Custom URL for a JSP
G. Initializing Parameters
H. Context Parameters
I. Welcome Pages
J. Loading Servlet on Startup
K. Defining Error Pages

**IX. Architecting Web Applications**
A. Model 1 architecture
B. Model 2 architecture
C. Model
D. View
E. Controller
F. Advantages of Model 2 Architecture

**X. Overview of Struts**
A. Struts Overview
B. Struts Components
C. Dynamic Web Project to Support Struts
D. Struts HTML input form
   1. HTML "form" tags
   2. The <html:form /> tag
   3. The <html:text /> tag
   4. The <html:submit /> tag
   5. The <html:reset /> tag
E. ActionServlet – The Controller
   1. ActionServlet
   2. ActionServlet Class
   3. How it Works
   4. Configuring the ActionServlet
F. The struts-config.xml
   1. The <form-beans> element
   2. The <action-mappings> element
   3. Other major elements
G. ActionForm – The View
   1. ActionForm Class
   2. Writing AddEmployeeForm class
   3. Entry in struts-config.xml
   4. ActionForm Methods
H. Action – The Controller
   1. Action class
   2. The execute () method
   3. Writing AddEmployeeAction class
   4. Action class – an example
   5. Entry in struts-config.xml
I. Validating Data in the ActionForm
   1. The validate() Method in ActionForm
   2. Entry in struts-comfig.xml

**XI. JSP and Servlet Filter**
A. Servlet and JSP Filters
B. What is a Filter?
C. Typical Uses of Filter
D. How Filters Work
E. Filter's Lifecycle
F. Writing the SimpleFilter
G. Deployment Descriptor
H. Multiple Patterns in Mapping

**XII. Using Annotations in Servlets**
A. Annotations
B. Allowable Annotations in a Servlet
C. PoostConstruct Annotation
D. PreDestroy Annotation
E. Resource Annotation
F. Environment Entry Variable
G. DataSource

# Java EE Web Application Development using Ganymede Eclipse

# Course Outline

**XIII. Overview of JEE 5 Architecture**

A. JEE 5 Architecture
B. JEE 5 Modules
C. HTTP Servlet
D. JavaServer Pages
E. Model View Controller
F. Struts
G. JSTL
H. EJB
I. J2EE Services
   1. Naming Service
   2. Database Access Service
   3. Transaction Service
   4. Messaging Service
   5. JavaMail Service
   6. Security Service