

The goal of Struts is to provide an open source framework for building Java Web applications. Struts encourages applications to be based on the Model 2 approach, which is a variation of the Model-View-Controller (MVC) design pattern. *Controller* components direct the flow of the application (e.g., servlets, JAXP). *Model* components are written in Java and perform business logic (e.g., JavaBeans, JDBC, EJB, JMS). *View* components display content to the end user (e.g., JSPs, HTML, XML). Struts can facilitate change and growth in web applications, and can also facilitate specialization of development team members.

This advanced three-day course will teach students how to use Java Struts as a framework to develop web applications that follow the Model/View/Controller design pattern. The topics cover the components of Struts that are available from the Jakarta project of the Apache Foundation. The course illustrates what the components provide and effective use of them.

Course Objectives:

- Describe how Struts fit in a Java application server environment.
- Use Struts to implement an MVC web application design.
- Configure a Struts application using struts-config.xml and web.xml.
- Build a JSP view using Struts and JSTL tags.
- Handle form validation, error processing, and logging in a Struts environment.
- Use the Struts Tiles facilities to make the look and feel of a web application flexible and easy to maintain.

Audience: Experienced Java Servlet and JSP developers who need to use Struts as a framework for MVC Web Application Development.

Prerequisites: *Java Programming, Java Web Programming, or Java2 Enterprise Edition.* Java Servlet and JSP programming experience, and a basic understanding of HTML and XML is required.

Number of Days: 3 days

- | | |
|---|--|
| <p>1. Course Introduction
 Course Objectives
 Course Overview
 Using the Workbook
 Suggested References</p> <p>2. Struts Overview
 What is Struts?
 Model 1 Design Pattern
 Model 2 / MVC Design Pattern
 Implementing MVC with a Framework
 The Struts Framework
 Basic Struts Components
 Struts Documentation
 A Struts-Based Application: Logon</p> | <p>3. Struts in a Simple Web Application
 Stars Information Application
 List Stars Flow
 Display Star Flow
 ActionServlet: the Controller
 struts-config.xml
 ActionForm: Form State
 The execute Method of StarsListAction
 The execute Method of StarsDisplayAction
 Directing Processing Flow with an ActionForward</p> |
|---|--|

Building a View with Tags
 Review: Flow through a Typical Struts-
 Based Application

4. The Controller

ActionServlet as a Controller
 RequestProcessor
 Developer Responsibilities
 Mapping
 Forwards

Lifecycle of an ActionForm
 ActionForm Considerations
 The reset Method
 The validate Method

5. Action and the Business Model

The execute Method of Action
 execute() Method Considerations
 Handling an Error
 Threading Considerations
 Some Best Practices for Action
 More Best Practices for Action

6. The View

Forwarding to a View
 Overview of Struts Tags
 Struts HTML Tags
 Form-Related Tags
 Dealing with URLs
 Using Error Tags
 Displaying Messages
 Struts Bean Tags
 Struts Logic Tags
 Some Struts View Best Practices

7. Internationalization

I18N and L10N
 Resource Bundles
 Java's MessageFormat Class
 Internationalization in Struts
 I18N with Struts Tags
 I18N with JSTL tags
 I18N within Java Code

8. Advanced Struts Features

Accessing Bean Properties
 DynaActionForm: A Configurable Form
 Indexed and Mapped Properties in a
 Form
 Using indexed="true"
 Preventing Duplicate Form Submits

Using ForwardAction and
 IncludeAction
 DispatchAction
 LookupDispatchAction
 Implementing a
 LookupDispatchAction

9. Handling Errors

Error Handling Options with
 Struts
 Documenting Errors with
 ActionMessage
 JSP Error Pages
 Declarative Java Exception
 Handling

Logging in Struts

10. Validation

Validator Overview
 Validator Requirements
 Configuring Validator Rules
 Struts Validators
 Configuring the Struts Validators
 Configuring Form Validation —
 global and formset
 Configuring Form Validation —
 form and field
 Configuring Form Validation —
 arg
 Configuring Form Validation —
 var
 Validation with Regular
 Expressions
 ValidatorForm verses
 ValidatorActionForm
 Implementing a Validator
 Method
 Other Validator Implications

11. Page Composition with Tiles

Tiles Overview
 Building a Tiles Template
 Basic Tiles Example
 Tiles Definitions
 Additional Options with
 Definitions
 Placing Definitions in a
 Configuration File
 Using the <put> Tag

Enabling the Tiles Plug-In
Using Tiles

12. Appendix – Nested Tags

Why Nested Tags?

Using Nested Tags

Parent and Root Tags