Web Development with Struts 2

Course Summary

Description

The goal of Struts2 is to make web development easier for the developer. To achieve this goal Struts2 provides features to reduce XML configuration files, utilizes annotations. Actions are POJOs which increases testability and reduces coupling in the framework. In this course you gain knowledge of the Struts 2 architecture and exercises will give you a solid knowledge of all components of the Struts 2 framework.

Topics

- Introduction to Struts 2 Framework
- Struts 2 Architecture
- Struts Configuration File
- Action Classes
- The Persistence Layer
- The Interceptors
- Type Conversion
- OGNL and Tags Library
- Validation Framework
- Internationalization
- Inversion of Control
- Advanced Topics

Audience

This course is designed for Java Developers.

Prerequisites

Students should be familiar with JSP and have basic knowledge of XML.

Duration

Four days

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

ProTech
protechtraining.com

"Charting the Course ...

... to Your Success!"

Web Development with Struts 2

Course Outline

I. Introduction to Struts 2 Framework
- A. Introduction
- B. What is a Framework?
- C. Struts 2 Overview
- D. Struts 2 Implementation of MVC
- E. Processing Request
- F. Model – Action
- G. View – Result
- H. Interceptors
- I. Exception Handling
- J. The ValueStack and ONGL
- K. Advantages of Struts 2
- L. Struts 2 vs. Struts 1

II. Struts 2 Architecture
- A. Two Ways of Declaring Architecture
- B. Declarative Architecture
- C. XML Based Declarative Architecture
- D. The struts.xml file
- E. The struts.xml file – include tag
- F. The struts.xml file – package tag
- G. Actions
- H. Actions –Single Result
- I. Actions –Multiple Results
- J. Request and Form Data
- K. Accessing Business Services
- L. Accessing Data from the Action
- M. Value Stack
- N. Example
- O. Exercise

III. Struts Configuration File
- A. Overview
- B. The struts.xml File (package element)
- C. Package Inheritance
- D. The struts.xml File (include element)
- E. The struts.xml File (action element)
- F. The struts.xml File (result element)
- G. The global-results Elements
- H. Results
- I. Namespace Configuration
- J. Namespace Example
- K. The interceptor-related Elements
- L. Exercise

IV. Action Classes
- A. Action
- B. What Does an Action Do?
- C. Encapsulates Interaction with the Model
- D. Provides Locus for Data Transfer
- E. Returns Control String for Result Routing
- F. Packaging Your Actions
- G. Implementing Action class
- H. ActionSupport Class
- I. ActionSupport Class validate method
- J. Using Resource Bundles
- K. Accessing Resources
- L. Aware Interface
- M. ServletContextAware Interface
- N. ServletRequestAware Interface
- O. SessionAware Interface
- P. ServletResponseAware Interface
- Q. Example
- R. Exercise

V. The Persistence Layer
- A. JDBC
  1. Overview of JDBC
  2. Connecting to the DataBase
  3. Submitting the SQL statements
  4. Retrieving and processing data
  5. DB_Action Class
  6. Data Access Object
  7. Exercise
- B. Hibernate
  1. Overview
  2. Hibernate
  3. Architecture Of Hibernate
  4. Code Example
  5. Creating Persistent Java Objects
  6. Mapping Persistent Object
  7. Hibernate Configuration File
  8. Inserting new record
  9. Querying the database
  10. Example
  11. Exercise
- C. Java Persistence API (JPA)
  1. Introducing JPA
  2. Hibernate And JPA
  3. Using Annotation

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

Web Development with Struts 2

Course Outline (cont'd)

4. Using Hibernate EntityManager
5. persistence.xml
6. Hibernate_JPA Project
7. Example

VI. The Interceptors

A. Overview
B. Invocation of an Action
C. ActionInvocation Class
D. Built-in Struts 2 interceptors
E. Configuring Interceptors (struts.xml)
F. Setting and Overriding Parameters
G. Interceptor Interface
H. Create Your Own Interceptor
I. Example
J. Exercise

VII. Type Conversion

A. Introduction
B. Build in Converters
C. Primitives And Wrapper Classes
D. Customizing Error Messages
E. Custom Type Converters
F. Configuring the framework
G. Example: Custom Type Converters
H. Exercise
I. Global Type Converters

VIII. OGNL and Tags Library

A. Introduction
B. OGNL and Struts 2
C. The Value Stack
D. The ActionContext
E. Expression Language
F. Reading Value Stack Object Properties
G. Selecting The Root Object For OGNL
H. ValueStack Details
I. An overview of Struts tags
J. Using OGNL to Set Attributes on Tags
K. The property tag
L. The set tag
M. bean Tag
N. UI Struts Tag Library
O. UI Input Tags
P. The Form Tag
Q. The textfield, password Tags
R. The submit Tag
S. The reset Tag
T. The checkbox Tag
U. Non UI (Generic) Tags
V. if Tag
W. iterator Tag

IX. Validation Framework

A. Overview
B. Using a Validator
C. Validator Configuration
D. Validator Configuration (param)
E. Strurs Build-in Validators
F. Date validator
G. double Validator
H. requiredstring Validator
I. Example
J. Exercise
K. Writing a Custom Validator
L. Example
M. Exercise

X. Internationalization

A. Introduction
B. Java Support for I18n
C. Struts 2 Support of i18n
D. Request_locale Parameter
E. Example
F. Exercise

XI. Inversion of Control

A. Introduction
B. Dependencies
C. Injecting Dependencies
D. Setter Injection
E. Problem Description
F. Adding Spring to Struts 2
G. Web.xml
H. applicationContext.xml
I. struts.xml After Integration
J. Example
K. Exercise

XII. Advanced Topics

A. Alternative Method Invocation
B. Wildcard Method
C. Dynamic Method Invocation

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically