

Introduction to JavaServer Faces (JSF) Using RAD

WT1553

This training course introduces the developer to the JSF architecture and provides the basis for planning, developing, and deploying Web based applications using the JSF framework. After taking this class, the developer will be able to quickly construct dynamic server-side web pages using JSF.

Audience

- Software designers, developers and programmers

Prerequisites

- Participants should already have a solid understanding of Java programming and understand the basics of XML. The course also assumes a basic understanding of HTML syntax and JavaServer Pages syntax. Understanding of Enterprise Java Beans (EJB) is also a plus.

Course Length

- Three days

Learning Objectives

- Become familiar with the JSF tags for the commonly used HTML components in a form.
- Learn how to write event handler methods in a managed bean.
- Learn how to register the event handler method with a GUI component.
- Write navigation rule in faces-config.xml.
- Control the sequence of page display based on certain business logic.
- Add very basic validation to the intro.jsp page
- Display objects in an ArrayList in a data table

Course Outline

WTE6

Introduction to JavaServer Faces

- What is JavaServer Faces (JSF)?
- Why Use JSF?
- Nature of a JSF Application
- The Sun Reference Implementation (RI)
- JSF and MVC
- Faces Servlet
- Managed Bean
- The View
- A Simple JSF Application
- The Input Form: form.jsp
- The Result: thanks.jsp
- The Controller: AddressBean
- Page Flow or Navigation Rule
- How Does the Application Work?
- Under the Covers: The Rendered Form HTML
- Under the Covers: The Layout Tree

Managed Bean

- What is a Managed Bean?
- Why Do You Need Managed Beans?
- Managed Bean As a Controller
- Registering a Managed Bean
- Lifecycle of a Managed Bean
- Initializing a Managed Bean
- What Scope to Use?
- Value Binding
- Property Data Conversion
- Advanced Property Types
- Component Binding
- Other Binding Types
- Basic Input Processing

- Advanced Input Processing
- JSF Expression Language
- Language Basics
- Use of Expression in JSF
- Basic Expression
- Property Access Expression
- Map Access Expression
- Complex Access Expression
- Predefined Objects
- Operators

User Interface Components

- JSF UI Components
- JSF Component Tag Libraries
- Using the Tag Libraries
- HTML Tag Library
- HTML Tag Library
- Identifying Components
- Component Identifier
- Client Identifier
- Form Input Field Names
- Labels and Images
- Create a Form
- Buttons
- Links
- Adding Parameters to Links
- Text Input
- Simple Check Box
- Check Box Group
- Check Box Group Example
- Radio Button
- List Boxes
- Dynamic List Box
- Example
- Another Example
- JSF Core Library
- JSF Core Library - Components
- JSF Component Model
- Base Class Hierarchy
- Useful Base Classes
- Useful HTML Classes

JSF Event Handling

- Objectives
- Request Processing Phases
- Action Request Processing Phases
- JSF Event & Listener Model
- JSF Event Classes
- Event Classes
- Event Classes: Hierarchy
- Action Event Handling
- Action Event: Example
- Action Listener
- Value Change Event
- Value Change Event Handler
- Listener Classes
- Listener Interfaces
- Listener Classes: Hierarchy
- Action Listener Class
- Value Change Listener Class
- Phase Event Listener
- Request Processing Lifecycle
- Phase Identifiers

JSF Navigation

- Objectives
- JSF Navigation
- Example Action Listener
- faces-config.xml File
- JSF Navigation
- Navigation Rules: Example
- Choosing Between Redirect & Direct Rendering
- Navigation: Scenarios

Validation

- Objectives
- Introduction
- Validation
- Standard Validators
- Checking for Required Values
- Standard Validation Error Messages
- Validation Process
- Sequence Diagram
- Immediate Components
- Accessing User Input
- Immediate Example

Advanced GUI Components

- Objectives
- HTML Panel
- Applying Styles

- Grouping Components
- Data Table
- Using a Data Table
- Data Table Model
- Using a Data Table
- Displaying a Header and Footer Row
- Applying Styles
- Adding Links to a Row
- Using Component Binding
- Handling Action
- Using the Data Table Model
- Handling Action

JSF Techniques

- A Typical Application
- The Create Form
- Create Action Method
- Listing Items
- Editing an Item
- Adding the Edit Link
- The Edit Form
- Update Item
- Things to Keep in Mind

Using Java Standard Template Library JSTL with JSF

- Objectives
- Introduction to JSTL
- Using JSTL with JSF
- The < c:out > Tag
- The < c:if > Tag
- The < c:choose > Tag
- The < c:forEach > Tag
- International Text Display Tags

Custom Components

- Custom Components
- Requirements To Develop Custom Components
- Packages Useful for Custom Component Developer
- javax.faces.component.UIComponent
- Decoding
- javax.faces.component.UIComponent
- UIComponentTag
- Implementing Custom Component Tags
- FacesContext
- javax.faces.application.Application
- javax.faces.context.ResponseWriter
- javax.faces.component.UIComponent
- javax.faces.component.EditableValueHolder
- Tag Library: TLD
- Faces configuration file: faces-config.xml
- faces-config.xml
- Sample.jsp: Example Showing the Relationship Between the JSF Artifacts
- External Renderer
- Render Defined in faces-config File
- Implementing the Renderer
- Event Listeners
- Using ActionListener in JSP
- ActionListener in the Java Code
- ActionListeners in JSF
- Create Input Form
- Input Form (Code)
- Update faces-config.xml
- ActionEvent Handling
- Summary

Data Conversion Model

- Conversion Overview
- Different Viewpoints of Data
- Process of Conversion
- Why Local Values?
- Types of Standard Converters
- Standard Converters with JSF
- Three Ways to Use Standard Converters
- Bind UI Component to Backing Bean Property
- Use converter Attribute on the UI Component
- Use < f:converter > Tag with ConverterId Attribute
- Usage of Standard Converters
- Output of Using Standard Converters
- Converter Attribute
- Standard Converters to Display Output
- Standard Converters: Screen Shot
- Conversion Errors
- Displaying Error Messages
- Conversion Tag
- Conversion Error: Display

- Conversion Error
- Standard Validators
- Validation Errors: Screen Shot
- Standard Validators
- Check Required Values
- Custom Converter
- Custom Converter
- Custom Converter
- Registering the Converter
- Using the Registered Converter
- Conversion Errors
- Summary

Appendix Faces Client Component

- Faces Client Overview
- Faces Client Overview
- Faces Client -Why
- Benefits of Faces Client Components
- Faces Framework Component
- Faces Framework Component
- Client Data
- Relational Record List
- Relational Record
- Page Data View
- Creating a Client Data Object
- Client Data View
- Binding client data to controls
- Screen shot - Client Data and Data Grid
- Summary