



Capstone Courseware, LLC

33 Boylston Street
Jamaica Plain, MA 02130

877-227-2477
capstonecourseware.com

115A. JavaServer Faces

Version 1.2.3

This course is an extension of Course 115, adding a two day-module on advanced techniques focused on custom component libraries, JavaScript, and Ajax. Proceeding from a foundation of Java, Servlets, and JSP, the course develops the best-practice concepts of MVC architecture and command-object encapsulation that propel the JSF architecture. Students create JSF Web applications by organizing their pages as JSF component trees, and their server-side code as JSF managed beans and controllers. They add data-conversion and validation logic and learn to work with JSF's data-table control.

Then, we change our perspective a bit and start thinking in terms of reusable component libraries -- using them and also developing them. Custom components prove to be the ideal delivery vehicle for JavaScript in JSF applications, and ultimately serve as the backbone of a robust Java/Ajax architecture. We work with two popular component libraries along the way -- Tomahawk and RichFaces -- and also get a look at Direct Web Remoting, or DWR.

A bridge module is also available that illustrates how JSF can work with EJBs and Java Persistence API entities; this brief presentation makes for a nice side trip during class time, for student groups who know they'll be working with JPA or EJB.

Prerequisites

General note: This course is intended primarily for experienced Java application developers. Page authors, component developers, and others who may have little or no Java experience (but perhaps are stronger on HTML and JSP) may well find this to be a valuable training experience, though without solid Java skills many of the coding exercises will be difficult to follow.

- Java programming experience is essential to understanding the JSF API as presented here -- Course 103 is excellent preparation.



115A. JavaServer Faces

Outline

- JSP page-authoring experience is required -- Course 112.
- Servlets programming experience is recommended but not required -- Course 110.
- Basic knowledge of XML will be helpful, as will any previous experience with HTML.





Learning Objectives

- Understand the purpose and scope of the JSF architecture
- Build Web applications using JSF's FacesServlet, faces-config.xml, and the JSF request/response lifecycle
- Use managed beans to encapsulate not only server-side form handling but also client-side presentation logic
- Implement control logic as JSF event listeners or action methods.
- Use validators and converters to implement a validation phase for a JSF application
- Use custom component libraries.
- Develop custom components for a variety of purposes, including composite components.
- Use custom components to encapsulate and deliver JavaScript content.
- Develop Ajax applications using JSF and DWR.
- Develop Ajax applications using RichFaces.

Timeline: 5 days.

IDE Support: Eclipse WTP 2.0

In addition to the primary lab files, an optional overlay is available that adds support for Eclipse WTP 2.0. Students can code, build, deploy, and test all exercises from within Eclipse, and take advantage of WTP's built-in editors, integrated debugging, and wizards for web applications, XML files, JSPs, and more. See also our orientation to Using Capstone's Eclipse Overlays, and please be advised that this is an optional feature; it is not a separate version of the course, and the course itself does not contain explicit Eclipse-specific lab instructions.





Module 1. JavaServer Faces

Chapter 1. Overview

- Java EE and Web Applications
- Perspectives: Servlets and JSP
- The Model/View/Controller Pattern
- Perspectives: MVC Frameworks
- The Command Pattern
- Perspectives: AWT and JFC
- JSF Value Proposition
- JSF Configuration

Chapter 2. Lifecycle and Page Navigation

- The JSF Request/Response Cycle
- Lifecycle Phases
- The FacesContext Class
- Who Does What
- View Selection
- Navigation Rules
- Outcomes

Chapter 3. User Interface Components

- The Composite View Pattern
- The UIComponent Class
- Behavioral Interfaces
- The Core and HTML Tag Libraries
- Relationship to CSS
- ID, Client ID, and Label
- The UISelectItem(s) Class
- Navigating the UI Tree

Chapter 4. Managed Beans

- JavaBeans and JSF
- Backing Beans
- The Factory Pattern
- Managed Beans
- Coarse-Grained Beans





- The Unified Expression Language
- Value Expressions
- Dependency Injection
- Bean Scopes

Chapter 5. Events and Controllers

- The Observer Pattern
- JSF Event Model
- Event Types and Timing
- Event Queueing
- ActionEvent and ActionListener
- Action Methods
- Connecting Controllers to Beans
- Dynamic Outcomes
- ValueChangeEvent and ValueChangeListener
- Limitations of FacesListeners

Chapter 6. Converters

- The Adapter Pattern
- The Converter Interface
- Standard Converters
- Working with Enumerated Types
- Timing of Conversion
- Custom Converters
- Configuring Converters

Chapter 7. Validators

- Validating Input
- The Validator Interface
- Standard Validators
- Producing Error Messages
- Message Keys
- Presenting Error Messages
- Custom Validators
- Validating Multiple Inputs
- Using a PhaseListener

Chapter 8. Data Tables

- Managing Tabular Data





115A. JavaServer Faces

Outline

- The JSF Data Table
- Columns
- Facets
- Limitations and Strategies
- Nesting Tables





Module 2. Custom Components

Chapter 1. Using Custom Components

- Component Libraries
- Deployment Model
- Apache Tomahawk

Chapter 2. Developing Custom Components

- The Components of a Component
- Custom Tags and CSS vs. Custom Components
- The UIComponent Class
- The Renderer Class
- Encoding and Decoding
- The UIComponentELTag Class
- Using the Default Renderer
- State-Saving Concerns

Chapter 3. Composites

- Composite Components
- JSF Composite vs. HTML Composite
- Composite vs. Shell
- Creating, Configuring, and Assembling

Chapter 4. Delivering JavaScript

- Client-Side Scripting
- HTML Events and the HTML DOM
- Encoding Scripts
- JSF and Script Libraries
- Encoding Scripts Once
- Encoding Scripts Uniquely

Chapter 5. Ajax Applications

- What is Ajax?
- Request Formats
- Means of Triggering HTTP Requests
- Asynchronous Response Handling
- Direct Web Remoting





Server-Side Strategies
Invoking Managed Beans

Chapter 6. Ajax Components

Server-Side Strategies: Another Look
Model/View/Controller for Ajax
JSF Lifecycle and Partial Page Rendering
RichFaces
Triggering Ajax Requests
Re-Rendering
Input Validation
Customizing Custom Components

Appendix A. Learning Resources

Appendix B. JSF Quick Reference

- JSF Custom Tags
- Packages and Classes
- JSF API: Finding Things
- JSF API: Controllers, Converters, and Validators

System Requirements

Hardware Requirements (Minimum)	500 MHz, 256 meg RAM, 500 meg disk space.
Hardware Requirements (Recommended)	1.5 GHz, 512 meg RAM, 1 gig disk space.
Operating System	Tested on Windows XP Professional. Course software should be viable on all systems which support a J2SE 5.0 JDK.
Network and Security	Limited privileges required -- please see our standard security requirements at http://capcourse.com/Guides/Security.html .
Software Requirements	All free downloadable tools.

