



Capstone Courseware, LLC

33 Boylston Street
Jamaica Plain, MA 02130

877-227-2477
capstonecourseware.com

117. The Spring Framework

Version 2.0.1

This course enables the experienced Java developer to use the Spring Application Framework to create simple and complex Web applications. Spring is a far-reaching framework that aims to facilitate all sorts of Java development, including every level of multi-tier distributed systems. Here we focus on the Core and MVC modules, with a lighter (but not dismissive) touch on persistence through DAO and ORM modules.

The Core module gives the developer declarative control over object creation and assembly; this is useful for any tier of any Java application. So is Spring's validation framework, and so we study these things in a mix of standalone (J2SE) applications and Web applications deployed to the Tomcat server/container. Then students build Web applications that use the Spring MVC framework to rationalize their designs into coherent request/response cycles. They use Spring command objects to manage HTML forms and their data, and connect these to the validation framework. We connect our applications to persistent stores and study the DAO and ORM modules, to better understand JDBC and Hibernate persistence models and declarative transaction control.

To serve a range of audiences who are interested in different parts of the Spring framework and who have different backgrounds, we've introduced new variants of this course. You might also consider Course 117A, which offers additional exercises and discussion to provide a more gradual, four-day timeline with a similar scope of coverage; and Course 117B, which is three days and has additional exercises on the Spring IoC container but de-emphasizes the Web module.

A bridge module is also available that illustrates how Spring's persistence module compliments the Hibernate ORM framework; this brief presentation can easily be added to the end of the class and works especially well to cap off a week of training using this course and our Hibernate course.



Prerequisites

- Java programming -- Course 103 is excellent preparation.
- Servlets programming -- Course 110.
- JSP -- Course 112.
- Basic knowledge of XML -- Course 501.





Learning Objectives

- Understand the scope, purpose, and architecture of Spring
- Use Spring's Inversion of Control to declare application components, rather than hard-coding their states and lifecycles
- Use Dependency Injection to further control object relationships from outside the Java code base
- Create validators for business objects, and associate them for application-level and unit-testing uses
- Build a Web application as a Spring DispatcherServlet and associated application context, with declared beans acting as controllers, command objects, and view resolvers
- Build and manage HTML forms with Spring command objects and custom tags
- Use Spring interceptors to implement horizontal features in the Web application
- Connect business objects to persistent stores using Spring's DAO and ORM modules

Timeline: 3 days.

For less experienced Java programmers who nonetheless meet the stated prerequisites, a 4-day timeline may be appropriate, to give some background on Java web applications and some breathing room around the later lab exercises.

IDE Support: Eclipse WTP 1.5

In addition to the primary lab files, an optional overlay is available that adds support for Eclipse WTP 1.5. Students can code, build, deploy, and test all exercises from within Eclipse, and take advantage of WTP's built-in editors 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.





Chapter 1. Overview

- Web Applications
- J2EE: The Good, The Bad, and the Ugly
- Enter the Framework
- Spring Modules
- Controlling Object Creation
- Web Applications
- Persistence Support
- Aspect-Oriented Programming
- Integrating Other Frameworks

Chapter 2. Core Techniques

- Component-Based Software
- JavaBeans, Reconsidered
- The Factory Pattern
- Inversion of Control
- XML View: Declaring Beans
- Java View: Using Beans
- Singletons and Prototypes
- Initializing Bean State

Chapter 3. The Business Tier

- Complex Systems
- Assembling Object Graphs
- Dependency Injection
- Single and Multiple Relationships
- Autowiring
- Bean Aliases
- Order of Instantiation
- Validation
- Nested Properties

Chapter 4. The Web Tier

- Servlets and JSPs: What's Missing
- The MVC Pattern
- The Front Controller Pattern
- DispatcherServlet
- A Request/Response Cycle
- The Strategy Pattern





- JavaBeans as Web Components
- Web Application Contexts
- Handler Mappings
- "Creating" a Model
- View Resolvers

Chapter 5. Controllers and Commands

- Working with Forms
- Command Objects
- The Template Method Pattern
- Command Controllers
- Data Binding
- MultiActionController
- Scope and Granularity of Command Objects

Chapter 6. Working with Forms

- Property Editors
- Validating Form Input
- Form Controllers
- AbstractFormController
- SimpleFormController
- Spring Custom Tags
- <form:form> and Friends
- <form:errors>
- Reporting Errors

Chapter 7. Refining the Handling Cycle

- The Intercepting Filter Pattern
- Exception Handling
- Interceptors
- The Decorator Pattern
- Context and Lifecycle
- Awareness Interfaces
- Support and Utility Classes
- "Death By XML"

Chapter 8. The Persistence Tier

- The DAO Pattern
- The DaoSupport Hierarchy





The DataAccessException Hierarchy
JDBC DAOs
JdbcTemplate and RowMapper
Object/Relational Mapping
Hibernate DAOs
Transaction Control
AOP vs. Annotations

Appendix A. Learning Resources

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.

