

CURRICULUM

HIBERNATE v3.0

» INTRODUCTION TO MAPPING OBJECTS TO RELATIONAL DATABASES

- Serialization 2
- XML 2
- Object-Oriented Database Systems
- Mapping

» INTRODUCTION TO HIBERNATE

- Hibernate Architecture
- Hibernate Configuration
- Hibernate's Support for Other Technologies
- Installing Hibernate
- A "Hello World" stand alone application
- A Servlet-Based Hibernate Application

» CREATING PERSISTENT CLASSES

- Mapping a Basic Java Class
- Mapping a Class with Binary Data
- Mapping a Serializable Class
- Mapping a Class with Data/Calendar Attributes
- Mapping a Read-Only Class
- Mapping Classes Using Versioning/Timestamps

» MAPPING INHERITANCE WITH JAVA CLASSES

- Table-Per-Class Hierarchy Mapping
- Table-Per-Subclass Hierarchy Mapping
- Table-Per-Concrete-Class Hierarchy Mapping
- Persisting Interfaces

» WORKING WITH COLLECTIONS

- Associations
- Lazy Initialization
- Mapping Maps/SortedMaps
- Mapping Set/SortedSets
- Mapping Lists
- Mapping Arrays
- Mapping a Bidirectional Association

- » **USING PERSISTENT OBJECTS**
 - Persisting Objects
 - Loading Data into an Object
 - Updating and Finding Objects

- » **SCALAR QUERIES AND HIBERNATE QUERY LANGUAGE**
 - Queries
 - Named Queries
 - SQL Queries
 - Hibernate Query Language

- » **HIBERNATE CACHING**
 - Setting Up a SessionFactory Cache
 - Using the Persistent Object Cache
 - Using the Query Cache
 - Setting Up EHCACHE

- » **HIBERNATE TRANSACTIONS AND LOCKING**
 - Configuration
 - Database Support
 - Using Transactions
 - The Transaction API
 - Transaction Example Using Oracle
 - Locking

- » **HIBERNATE AND XDOCLET**
 - Introduction to XDoclet
 - Using XDoclet with Hibernate
 - Hibernate XDoclet Tags and Description

- » **HIBERNATE AND ECLIPSE**
 - Hibernate
 - Hiberclipse
 - Hibernate Synchronizer

- » **INTEGRATION OF STRUTS 2, HIBERNATE AND SPRING**
 - Struts and Hibernate
 - Spring and Hibernate
 - Struts, Hibernate and Spring

SPRING v2.5

- » **What is Spring**
 - Spring modules
 - Understanding dependency injection
 - Applying aspect-oriented programming

- » **Basic bean wiring**
 - Containing your beans
 - Creating beans
 - Injecting into bean properties
 - Autowiring
 - Controlling bean creation

- » **Advanced bean wiring**
 - Declaring parent and child beans
 - Applying method injection
 - Injecting non-Spring beans
 - Registering custom property editors

- » **Advising beans**
 - Introducing AOP
 - Creating classic Spring aspects
 - Creating advice
 - Defining pointcuts and advisors
 - Using ProxyFactoryBean
 - Autoproxying

 - » **Hitting the database**
 - Learning Spring's data access philosophy
 - Configuring a data source
 - Using JDBC with Spring
 - Working with JDBC templates
 - Using Spring's DAO support classes for JDBC
 - Integrating Hibernate with Spring
 - Caching

 - » **Spring and POJO-based remote services**
 - An overview of Spring remoting
 - Working with RMI
 - Remoting with Hessian and Burlap
 - Using Spring's HttpInvoker
 - Spring and web services

 - » **Building contract-first web services in Spring**
 - Introducing Spring-WS
 - Defining the contract (first!)
 - Handling messages with service endpoints
 - Wiring it all together
 - Consuming Spring-WS web services

 - » **Spring and Enterprise JavaBeans**
 - Wiring EJBs in Spring
 - Spring and EJB3

 - » **Handling web requests**
 - Getting started with Spring MVC
 - Mapping requests to controllers
 - Handling requests with controllers
 - Handling exceptions

 - » **Integrating with other web frameworks**
 - Using Spring with Struts
 - Registering the Spring plug-in with Struts
 - Writing Springaware Struts actions
 - Delegating to Spring-configured actions
 - What about Struts 2?
 - Working Spring into WebWork 2/Struts 2
-