Java EE: Integrating Hibernate and Spring 3.0

Meet the expert: Greg Matus combines his practical programming skills and mentoring excellence to develop the highest quality educational programs available. His technical focus includes developing programs and projects focused around advanced application development, performance, and tuning skills and technologies. His specialties include Java, advanced J2EE / JEE, AJAX, XML, Web Services, JSP, SQL, JDBC and Database topics, Oracle, Eclipse, WebLogic, and more.

Prerequisites: The students should be familiar with object-oriented principals and have a solid practical understanding of polymorphism in Java. They must also have a working knowledge of Eclipse IDE for JavaEE.

Runtime: 11:24:05

Course description: As data models get more complex it becomes harder for programmers to create applications to access that data. Enter Java Hibernate. Hibernate is a middle-tier developer framework for Object to Relational Mapping (ORM). A Hibernate tier is used as a go-between from the data tier to the client tier thus creating an automatic translation between the Object Oriented model and the required relational data model. This course will take the next step by showing you the higher functions of Hibernate like Mapping Inheritance, Annotations and Composite keys. You will then move on to Transactions and Querying, then Spring Hibernate Integration foundation and end with using Hibernate in Spring.

Course outline:

Mapping Inheritance
• Introduction
• Inheritance Mapping
• Patterns for Mapping Inheritance
• Single Table Inheritance
• Class Table Inheritance
• Concrete Table Inheritance
• Inheritance and Proxy Caveats
• Summary

Mapping Inheritance B
• Introduction
• Demo: Mapping with Code
• Demo: Mapping Subclasses
• Demo: Running Test Client
• Summary

Hibernate Annotations
• Introduction
• Annotations Overview
• Hibernate Annotations Overview
• Setting Up to Use Annotations
• Annotation Drawbacks Benefits
• Getting to Hibernate Annotations
• Declaring Persistence
• Generating Primary Keys
• Primary Key Annotation Examples
• @SequenceGenerator
• @TableGenerator
• Compound Primary Keys
• Compound-Primary-Key Class

• Declaring Compund PK Class
• Nested Classes
• Compound Primary Key Examples
• Client-Code for Compound PK
• Mapping Relationships
• Demo: Annotations
• Demo: Testing
• Summary

Annotations Composite Keys
• Introduction
• Demo: Mapping
• Demo: Test Client
• Summary

Reading Updating and Deleting
• Introduction
• Session Scope
• Important Session Methods
• States
• Transient to Persistent
• Hibernated to Persistent
• Detached to Persistent
• Summary

Reading Updating and Deleting
• Introduction
• Demo: Session-per-Request
• Demo: InvocationHandler
• Demo: Exceptions
• Demo: Member DAO Class
• Demo: Run Client
• Summary

Transactions
• Introduction

• Transactions
• Framework-Managed Transactions
• Isolation Levels
• Demarcating Transactions
• Managed Environments Need Config.
• Demarcating Transactions
• Concurrency
• Optimistic Locking-Versioning
• Pessimistic Locking
• Pessimistic Locking Levels
• Pessimistic Locking Example
• Demo: Hibernate Transactions
• Demo: Effects on Objects
• Demo: Running Test
• Summary

Querying For Objects
• Introduction
• Hibernate Querying
• Hibernate Type Implementations
• Hibernate Query Language
• The from Clause
• Associations and Joins
• HQL select Clause
• HQL where Clause
• HQL Aggregate Functions
• Expressions
• order by and group by Clauses
• Pagination

• The Criteria API
•
  org.hibernate.criterion.Restrictions

• org.hibernate.criterion.Order
•
  org.hibernate.criterion.Projections

• Detached Queries
• Summary

Querying for Objects B
• Introduction
• Demo: Basic Querying
• Demo: Run Test Query Client
• Demo: Querying Using Parameter
• Demo: Using Criteria Object
• Demo: Addresses Using SQL
• Summary

Spring Hibernate Integration
• Introduction
• Data Access Support
• DAO Implementations
• Transaction Support
• Isolation Level Concepts
• Isolation Level Constants
• Propagation Behavior Constants
• Transaction Management
• Transaction Configuration
• Programmatic Transaction
• Spring - Declarative Transactions
• Benefits of Declarative Transactions
• Forms of Declarative Transactions
• Using the Spring Config File

Java EE: Integrating Hibernate and Spring 3.0

- The Java Code is Simple
- Configuring the Transaction
- Using @Transactional
  Annotation
- Declarative Annotations
- Summary

Using Hibernate in Spring
- Introduction
- Intro
- Benefits of Using Spring
- Configuring Hibernate in Spring
- Using Hibernate in Spring
- Transaction Management
- Open Session in View
- Demo: Configuring Hibernate
- Demo: Test and Run
- Demo: Recap
- Summary