

## Developing J2EE Applications Using Hibernate Course Outline

### 1. Introduction to Hibernate

- What is Hibernate?
- Relational Database Technology
- Accessing Relational Databases from Java
- Domain Models
- Comparing Object and Relational Models
- Identity Modeling
- Granularity of Objects and Tables
- Association Modeling
- Navigation
- Inheritance
- What is Object/Relational Mapping?
- Advantages of using an ORM Tool
- Architecture of Hibernate
- Hibernate API
- Extension Interfaces
- Hibernate Configuration
- Hibernate Mappings
- Hibernate Properties
- Hibernate XML Configuration Files
- Configuration in a Non-Managed Environment
- Configuration in a Managed Environment
- Binding SessionFactory to JNDI
- A Simple Example
- Alternatives to Hibernate
- EJB Entity Beans
- Java Persistence API
- Hand-Coded Persistence Using JDBC
- Object-Oriented Databases
- Other ORM Tools

### 2. Basic Mappings

- Creating the Domain Model
- Domain Model
- Domain Objects
- Hibernate Requirements
- Object Identity
- Primary Keys
- Natural Keys vs. Surrogate Keys
- Identifier Property
- Identifier Generators
- Hibernate Managed Identifiers
- Mapping Metadata

- Mapping Properties
- Property Access Strategies
- Derived Properties
- Preventing Insert/Update
- Using Database Schemas
- Quoted SQL Identifiers
- Specifying Java Packages
- Entity and Value Types
- Components
- Bi-Directional Components
- Other Component Details
- Association Mappings
- Many-to-One Association
- One-to-Many Association
- Inheritance Mapping Strategies
- Table per Concrete Class
- Table per Subclass
- Table per Class Hierarchy

### 3. Manipulating Persistent Objects

- Object States
- Transient Objects
- Persistent Objects
- Detached Objects
- Object Identity
- Persisting a Transient Object
- Retrieving and Updating a Persistent Object
- Deleting a Persistent Object
- Updating a Detached Object
- Transitive Persistence
- Using save-update
- Differentiating Between Transient and Detached Objects
- HQL Queries
- Parameter Binding
- Pagination
- Named Queries
- Polymorphic Queries
- Restriction
- String Comparison
- Logical Operators
- Ordering the Results
- Join Queries
- Implicit Joins
- Where Clause Joins
- Criteria Queries
- Polymorphic Criteria Queries
- Restricting Criteria Queries
- Comparing Strings in Criteria Queries
- Using Logical Operators in Criteria Queries

- Ordering Results in Criteria Queries
- Using Joins in Criteria Queries
- Using Dot Notation to Get Component Properties

#### 4. Advanced Mappings

- Topics
- One-to-One Associations
- Many-to-Many Associations
- Polymorphic Associations
- Entity vs. Value Types
- Built-in Mapping Types
- Using Mapping Types
- Custom Mapping Types
- Collections of Value Types
- Set of Value Types
- Bag of Value Types
- List of Value Types
- Map of Value Types
- Collections of Components
- Sorting Collections

#### 5. Transactions, Concurrency and Performance

- Database Transactions
- JDBC and JTA Transactions
- Hibernate Transactions
- Flushing the Hibernate Session
- Isolation Issues
- Isolation Levels
- Configuring the Isolation Level
- Pessimistic Locking
- Application Transaction Isolation
- Managed Versioning
- Using Existing Properties to Implement Optimistic Locking
- Caching
- Hibernate Cache Architecture
- The First-Level Cache
- The Second-Level Cache
- Cache Concurrency Strategies
- Enabling the Second-Level Cache
- Cache Regions
- Specifying a Cache Provider
- Support for Cache Providers
- Explicitly Evicting Objects From the Second-Level Cache
- Fetching Strategies
- Fetching Strategies in Mappings Single Point Associations
- Fetching Strategies in Mappings Collections
- Fetching Strategies in Mappings Batches
- Explicit Initialization of Lazily Loaded Objects

- Fetching Strategies in HQL Queries
- Fetching Strategies in Criteria Queries
- The N+1 Selects Issue
- Report Queries Projections
- Report Queries Aggregation
- Report Queries Grouping

## 6. Tools and Design Techniques

- Generating the Database Schema
- Generating the Domain Model
- Generating the Database Schema
- Using Middlegen
- Using XDoclet
- The Open Session in View Pattern
- Using a Natural Key
- Using a Composite Key
- Dealing with Database Triggers