

Hibernate 4- An ORM Tool

Workshop Details:

Duration:	3 Days
Objective:	<p>This course is a upgrade on OR/M for developers from Core Java platform or from non-java platform.</p> <p>On completion of this course, developers will be able to understand:</p> <ul style="list-style-type: none"> ▪ What is JDBC, JDBC Wrapper and ORM tools ▪ Why to use ORM tool? ▪ What is the difference between Hibernate & JPA ▪ The Hibernate architecture and how to do the configuration ▪ How to use Service Registry for building session factory ▪ How to Use Apache Maven for Resolving Hibernate Dependencies ▪ How Mapping is done using Annotations ▪ What is the Object lifecycle ▪ What are the different Hibernate Inheritance strategies ▪ What are the different Hibernate Association strategies ▪ Who to Query a database using HQL ▪ What are the different types of queries
Participants' Entry Profile:	<p>Must have thorough knowledge of Core Java and should be aware of JDBC (Java Database Connectivity). Should have knowledge of RDBMS concepts and SQL-Joins. Prior Knowledge of topics such as JPA, Spring JDBC will be advantageous.</p>
Training Methodology:	<p>The workshop will follow Synergetics methodology of</p> <ul style="list-style-type: none"> Concept Visualization Active Experimentation Application Development <p>The workshop will be 100% Hands-On with each participant having access to system during the session</p>

Setup Requirements

<p>Hardware and Software Requirements:</p>	<p>Participant's as well as Trainer's Machine are required to have :</p> <p>Hardware</p> <ul style="list-style-type: none"> Intel Core i3 2.2 GHz CD Rom Drive 80 GB HDD LCD Color Monitor 2 GB RAM LAN Connectivity <p>Software</p> <ul style="list-style-type: none"> Windows XP or 7 Internet Explorer 10 or above/Chrome/Firefox Java SDK 1.7.x Tomcat 7.0/GlassFish 3.2.x/JBoss 7+ Hibernate 4.3.x All Jar JDBC Driver for Oracle (ojdbc6.jar) Oracle 10g or higher/SQL Server 2008 onwards Eclipse 4.0.x onwards/Netbeans 7.4 or higher Jboss tools from eclipse market place <ul style="list-style-type: none"> • The installable must match in bits of the machine. • A disk space shared with participants and trainer should be provided for trainer to share training stuff with participants. • Internet connection to fetch maven dependencies .
<p>Training Lab Requirements:</p>	<p>Whiteboard 6 feet by 4 feet (minimum) Whiteboard markers – Red, Blue, Green, Black Video Projector (1024 X 768 resolutions)</p>

Course Content

Day 1	<ul style="list-style-type: none"> ➤ Module 1: Quick overview of Object Relational Mapping, Hibernate <ul style="list-style-type: none"> ■ JDBC and JDBC Wrapper ■ Gaps between RDBMS and OOPs paradigm ■ Why ORM? What concerns it addresses? ■ Comparison ORM with JDBC, Spring JDBC and JPA ■ Features of Hibernate ■ Hibernate Architecture ■ Configuration Tags and Significance ■ Hibernate Configuration Properties ■ Configuration, Session Factory and Session ■ Hiber 4: ServiceRegistry, StandardServiceRegistryBuilder ➤ Module 2: Persistent Classes configuration in further detail <ul style="list-style-type: none"> ■ Understanding object states ■ Transient, Persistent, Detached ■ Object transition ■ Understanding HBM tags ■ Basic OR Mapping ■ Mapping Declaration ■ Mapping using Annotations ■ CRUD Operations ■ Persistence using Pojo ■ Accessing fields ■ Introducing constraints like not-null ■ Making column/class immutable
Day 2	<ul style="list-style-type: none"> ➤ Module 3: Working on keys <ul style="list-style-type: none"> ■ Auto key generators ■ Handling concatenated keys ■ Hibernate Data types ➤ Module 4: Working with collections <ul style="list-style-type: none"> ■ Persistent Collections ■ Collection Mapping <ul style="list-style-type: none"> ○ Set, Bag, List ○ Lazy Initialization ➤ Module 5: Filling gaps of Inheritance between OOPs and RDBMS <ul style="list-style-type: none"> ■ Mapping table/s with Inheritance object structure ■ Table with discriminator ■ Table per concrete class ■ Table with sub-class joins
Day 3	<ul style="list-style-type: none"> ➤ Module 6: Filling gaps of Association between OOPs and RDBMS <ul style="list-style-type: none"> ■ Uni and Bidirectional Association ■ Mapping- 1-1, 1-M, M-M ■ The (n+1) problem ➤ Module 7: Hibernate Query Language(HQL) <ul style="list-style-type: none"> ■ The 'from' and 'where' clauses ■ Aggregate Functions

- Orderby, GroupBy, Having
- Associations and Joins
- Object Graph Navigation
- Paging

- **Module 8: Criteria Queries**
 - Criteria Instance
 - Adding restrictions- SimpleExpression
 - Ordering and paging
 - Handling Projections

- **Module 9: Join Queries**
 - Inner and Outer Join
 - Fetch joins
 - Solution for (n+1) problem
 - Implicit and Theta style joins