Java Persistence API – Course Outline

# Duration

3 days

# Objectives

This course provides thorough coverage of Java Persistence Architecture and API to create and manage Object Relational model along with Hibernate as JPA provider.

Note: *Tomcat 7 / Weblogic 11 will be used as Server for the labs*

# Audience

This course is for people who wanted to build domain model and data tier in Java Enterprise Application using Object Relational mapping / modeling.

# Pre-requisite

Participants are expected to possess following pre-requisite skills

* Good in Core Java Programming and all the technologies that are part of Core Java
* Good hands-on experience with Eclipse IDE (Setting up Projects, build path)
* Used build tools like ANT or MAVEN
* Good in Java 1.5 / 1.6 features like Annotations

# Outline - Detailed

## Day 1

### Background

1. Traditional or Classical JDBC way
2. Need for Object – Relational mapping
3. Comparing Object oriented world with Relational world
4. Evolution and current state of ORM
5. **Hands-on:** JDBC way with a simple schema

### Entities and Object Relational Mapping

1. Entity Class
2. Persistent Fields and Properties
3. Configuring Hibernate as JPA provider
4. Quiz on Entity Classes
5. **Hands-on:** Entity Classes with ORM mapping creating DB schema with code-first approach (DB schema created in database using ORM mapping)
6. Entity Relationships
7. Relationship mapping defaults
8. Metadata for Object Relational mapping
9. Quiz on Entity class relationships and ORM Mapping
10. **Hands-on:** Entity class relationships and ORM Mapping generating DB schema with relevant integrity constraints

### Entity Manager and Persistence Context API

1. Introducing Persistence Context
2. Overview of EntityManager API
3. **Hands-on:** Perform simple CRUD operations on entities using EntityManager

## Day 2

### Creating BaseDao framework using Java Generics

**Hands-on**

1. Create base entities
2. Create BaseDao with EntityManager API to do CRUD operations
3. Add EntityListener with callback methods to be invoked during CRUD operations
4. Test BaseDao for CRUD operations

### State transition of an Entity

1. Entity Manager & Entity Instance’s Life Cycle
2. Persistence Context Lifetime
3. Transaction context / Transaction
4. **Hands-on:** Operations in EntityManager and corresponding state transitions

### Advanced Entities

1. Embeddable classes
2. Primary Keys and generators
3. Inheritance
4. Mapping Collections – Unidirectional and Bidirectional
5. Configuring Eager and Lazy loading
6. **Hands-on:** Modeling complex entity structure with many-to-many mapping, collections and performing CRUD operations on this model

## Day 3

### JPA Query Language

1. Query APIs
2. Statement Types, Abstract Schema types and Query Domains
3. SELECT, FROM, WHERE Clauses
4. Conditional Expressions
5. GROUP BY, Having, ORDER BY
6. Bulk Update and Delete Operations
7. Null values, Equality and Comparison Semantics
8. Quiz on Query API / Language
9. **Hands-on:** Implement reports using Query API / Language

### Native Queries and Named Queries

**Hands-on:**

1. Writing Native SQL queries
2. Accessing objects returned by native SQL
3. Creating Named Queries
4. Invoking named queries with parameters

### Criteria API

1. Meta model API
2. Criteria API Interfaces
3. Constructing Criteria Queries
4. Constructing Strongly-typed Queries
5. Use of Criteria API with Strings to Reference Attributes
6. **Hands-on:** Implement search using Criteria Queries

### Interceptors and Deployment descriptors

1. Controlling Transactions
2. Container and Application managed Persistence Contexts
3. Cache and PersistenceUnitUtil Interface
4. Requirements on the Container and Runtime Contracts
5. Entity Packaging
6. Interceptor Life Cycle
7. Business Method, Timeout Method, Lifecycle Event Callback interceptors
8. Interceptors in Deployment Descriptor
9. Quiz on Interceptors
10. Hands-on Interceptors
11. Goals for Exception handling
12. Application Exceptions

# Hardware & Network Requirements

* Minimum Core 2 Duo with 2 GB RAM

# Software Requirements

* Windows XP Professional or higher or Linux
* JDK 1.7, Eclipse Kepler or above
* Weblogic 11 Application Server
* Tomcat 7 web server
* MySQL 5.5.+ database