

Introduction to Data Persistence with JPA

Description

The Java Persistence API Training Course is a step-by-step introduction to building applications using the new Java persistence framework. JPA Stands for Java Persistence API, also known as EJB3 Persistence or JEE 5 Persistence. It is a lightweight framework for creating Java database applications.

It lets you develop persistent classes and POJO objects to the relational database using following common Java idiom such as – association, inheritance, polymorphism, composition and the Java collections framework.

Classroom Registration Price (CHF)

2300

Virtual Classroom Registration Price (CHF)

2150

Course Content

Module 1: Introduction to Java Persistence API (JPA)

- Lesson 1: Overview
 - Persistence Layers, Object-Relational Mapping (ORM), JDBC
 - JPA Overview
 - Lesson 2: Mapping with JPA
 - Entities and @Entity, ids and @Id,
 - Generated Id Values
 - Basic Mapping Types
- Lesson 3: Persistence Unit and EntityManager
 - Persisting to the DB, the EntityManager API
 - Persistence Units, Config, Persistence Context
 - Retrieving Persistent Entities with find()
- Lesson 4: More About Mappings
 - Default Mappings, @Basic, @Column
 - Field vs. Property Access
 - Temporal (Date/Time) Mappings
 - Lesson 5: Logging Options (Provider based)

Module 2: Updates and Queries

- Lesson 1: Inserting and Updating – Persisting new Entities, Updating an Instance, Removing an Instance
- Lesson 2: Querying and JPQL
 - Entity Based Queries, SELECT ,WHERE
 - Query Interface, Executing Queries, Generic Queries (JPA 2)
 - JPQL Operators, Expressions, and Parameters
 - Named Queries
- Lesson 3: Additional Query Capabilities
 - Projection query, Ordering, Aggregate Query, Build Update and Delete
- Lesson 4: Embedded Objects
 - @Embeddable, @Embedded
 - Defining and using Embedded Objects
 - Compound Primary Keys – @EmbeddedID, @IDClass, Defining Compound Keys

Module 3: The Persistence Lifecycle

- Lesson 1: Transaction Overview and Transactions in JPA
 - Transaction Overview
 - EntityTransaction API (including JTA and resource-local EntityManager)
- Lesson 2: The Persistence Lifecycle
 - JPA Entity States (New, Managed, Detached, Removed), and Entity State Diagram
 - Persistence Context – Lifespan, Propagation
 - Synchronization to the DB
- Lesson 3: Versioning and Optimistic Locking
 - Overview, Detached Instances
 - Versioning, @Version, Optimistic Locking
- Lesson 4: Lifecycle Callbacks
 - @PrePersist, @PostPersist, etc.
 - Entity Listeners, @EntityListeners

Module 4: Entity Relationships

- Lesson 1: Relationships Overview: Object Relationships, Participants, Roles, Directionality, Cardinality
- Lesson 2: Relationship Mapping
 - Mapping Overview (1-1, 1-N, N-1, N-N)
 - Unidirectional and Bidirectional
 - @ManyToOne, @OneToMany, Table Structures
 - Collection Types (List, Set, etc)
 - Cascading Over Relationships (including orphanRemoval – JPA 2)
 - @ManyToMany, @OneToOne
 - Lazy and Eager Loading
 - Queries Across Relationships (Inner Joins, Outer Joins, Fetch Joins)
- Lesson 3: Entity Inheritance Mapping
- Lesson 4: Overview
 - Single Table Mapping
 - Joined (Table per Subclass) Mapping
 - Table per Concrete Class Mapping
 - Pros and Cons
- Lesson 5: Element Collections (JPA 2)
 - Overview, Collections of Value Objects, @ElementCollection, @CollectionTable
 - Using Element Collections
 - Collections of Embeddables

Module 5: The Criteria API (JPA 2)

- Lesson 1: Overview of the Criteria API
- Lesson 2: Path Expressions, Building Queries (CriteriaBuilder, CriteriaQuery, Subquery, Predicate, Expression, Order, Selection, Join)
- Lesson 3: Executing Queries and Accessing Results

Module 6: Additional JPA Capabilities

- Lesson 1: XML Mapping Files
- Lesson 2: Bean Validation (JPA 2)
- Lesson 3: Best Practices
- Lesson 4: Primary Keys, Named Queries, Lazy/Eager Loading, Transactional Semantics, Encapsulation, Report Queries

Module 7: Integration

- Lesson 1: Data Access Objects (DAO) and Java SE Integration (Optional):
 - DAO Overview
 - JpaUtil Class for EntityManager management in Java SE
 - Lifecycle Considerations
- Lesson 2: Integration with EJB (Optional):
 - Using JPA with Session Beans
 - Container Managed (Injected) Entity Manger
 - JTA Transactions and Lifecycle Considerations
 - Extended Persistence Contexts
- Lesson 3: Using JPA with Java Web Apps
 - Using EntityManager in Web apps – request scoping
 - Lazy Loading – Open EntityManager in View Pattern
- Lesson 4: Integration with Spring (Optional)
 - Injection of EntityManager, EntityManagerFactory
 - LocalEntityManagerFactoryBean
 - JPA/Spring Based DAO

Documentation

- Digital courseware included

Participant profiles

- Beginner developers who want to develop POJO's based-applications interacting with the relational database

Prerequisites

- Intermediate knowledge of Core Java programming
- Creating basic web applications using Servlet & JSP
- Some familiarity with SQL, relational databases, and the Java Database Connectivity (JDBC) interfaces will be helpful

Objectives

- Understand how to persist your class POJO's with relational database using JPA Object-Relational mapping technology
- Be confident for implementing and maintaining Object-Relational persistence in your JPA application, including all necessary client-side and server-side programming

Niveau

Intermédiaire

Duration (in Days)

3

Reference

JPA
