Course : JPA, data persistence in Java

Practical course - 3d - 21h00 - Ref. JPA
Price : 1930 CHF E.T.

Inspired by the Hibernate and Toplink frameworks, JPA has become the standard for managing the persistence of Java objects associated with a database. In this training course, you will implement the concept of object-relational mapping, guaranteeing greater portability for your Java EE applications.

Teaching objectives

At the end of the training, the participant will be able to:

- Mapping Java objects to relational tables
- Create, update and delete persistent objects
- Master the JPQL query language
- Manage transactions

Intended audience
Developers, software architects and project managers.

Prerequisites
Good knowledge of Java and the JDBC API.

Practical details

Hands-on work
Exercises are carried out using the JPA implementation of Hibernate, Eclipse, MySQL/MariaDB and Tomcat.

Course schedule

1 Java and JPA persistence techniques
- Different persistence mechanisms: Java APIs and frameworks.
- The Java Persistence API (JPA) solution.
- Integrating JPA into a Java EE application.

PARTICIPANTS
Developers, software architects and project managers.

PREREQUISITES
Good knowledge of Java and the JDBC API.

TRAINER QUALIFICATIONS
The experts leading the training are specialists in the covered subjects. They have been approved by our instructional teams for both their professional knowledge and their teaching ability, for each course they teach. They have at least five to ten years of experience in their field and hold (or have held) decision-making positions in companies.

ASSESSMENT TERMS
The trainer evaluates each participant's academic progress throughout the training using multiple choice, scenarios, hands-on work and more.
Participants also complete a placement test before and after the course to measure the skills they've developed.

(2) Developing a persistent class

- Code the persistent class.
- Object/relational mapping.
- Configure and start the JPA engine.
- Perform a JPQL query.
- Save a persistent object.

Hands-on work
Mapping a class. JPQL query execution.

(3) Object/relational mapping with JPA

- Context and objectives of an ORM.
- Principle of persistent class development.
- Use annotations to configure Object/Relational mapping.
- Mapping classes and associations.
- Mapping strategy for inheritance.

Hands-on work
Choosing an identifier generator. Implementing one-to-one, one-to-many and inheritance relationships.

(4) Handling persistent objects

- Different reading techniques.
- Loading strategies.
- Principle of lazy loading.
- CRUD (Create/Read/Update/Delete) operations.
- Lifecycle of persistent objects.
- Synchronization with the database.

Hands-on work
Create, update and delete persistent objects.

(5) Advanced mapping

- Compound primary key, multitable mapping.
- Control INSERT and UPDATE queries.
- List, map and many-to-many associations.

Hands-on work
Mapping a secondary table. Implementing a many-to-many association.

(6) The JPQL language

- Query requests.
- Operations on character strings and temporal data.
- Internal, external and patch joints.
- Principle of subqueries.
- Set queries.

Hands-on work
Perform an optimized JPQL query.

7 Competing transactions and access

- Reminder of transaction properties.
- Transaction management with JPA.
- Integration into Web and EJB applications.
- Pessimistic and optimistic locks.

Hands-on work
Implementation of transactional management.