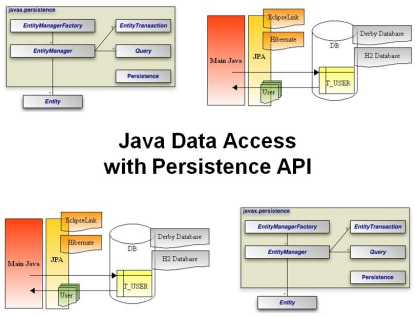
Java Data Access with JPA

Duration: 3 days **Price: € 1850**

Open Schedule



Java Data Access with Persistence API

Audience Course Java Data Access

Experienced Java developers who want to learn how to use the Java Persistence API for accessing data in databases.

Prerequisites Java Data Access with Persistence API

Experience with Java programming and object orientation is required to participate in this course. Knowledge of database structures and SQL is beneficial for a good understanding.

Realization Training Course Java Data Access

The concepts are treated on the basis of presentation slides and demos. The theory is interspersed with exercises. All topics in the Java EE Persistence API Developer Certified Expert Exam (1Z0-898) will be discussed. The course material is in English. The course times are from 9.30 up and to 16.30.

Certification Java Data Access with Persistence API

Participants receive an official certificate Java Data Access with Persistence API after successful completion of the course.

Content Course Java Data Access with JPA

In the course Java Data Access with JPA Object Relational Mapping with the Java Persistence API 2.x is on the agenda.

JPA Architecture

After an overview of the data access capabilities in Java, including JDBC, and the challenge they face, the basic concepts and architecture of the JPA is explained. The role of the Entity Manager and the persistence.xml configuration file is discussed.

Entities

The next topic covers the concept of a JPA Entity, the lifecycle of Entities and how Entity classes are mapped to database tables. Also the various states Entities can have in relation to the database such as new, persistent, detached and removed are discussed in this respect as is the concept of merging.

Annotations

Furthermore the use of annotations and XML mapping files and the role the various properties and attributes play is explained.

JPA Queries

The various key generation strategies are discussed as well as the mapping of association relationships and the mapping of inheritance relationships between Entities. Next the possibilities of the JPA Query language, JPQL, are covered and the uses of JPA criteria and native SQL queries.

Callbacks

The function and operation of Entity callbacks that are called immediately before and after the execution of a persistence operation is explained and the alternative use of Entity Listener classes as well.

Interceptors

Part of the course program is also the treatment of interceptors which are used for crosscutting concerns like logging and security. Finally, attention is paid to the use of JPA in a Java EE application, how to interact with EJBs and the method of packaging JPA entities.

Transactions

Finally JPA transactions are addressed in both a desktop environment and a Java EE environment.

SpiralTrain BV
Standerdmolen 10, 2e verdieping
3995 AA Houten

info@spiraltrain.nl
www.spiraltrain.nl
Tel.: +31 (0) 30 – 737 0661

Locations
Houten, Amsterdam, Rotterdam, Eindhoven, Zwolle, Online

Modules Course Java Data Access with JPA

Module 1 : Intro Java Persistence	Module 2 : Persistence API	Module 3 : Mapping Persistent Objects
Java Persistence Traditional Persistence Transparent Persistence Persistence Technologies Direct File I/O Serialization Java Database Connectivity JDBC Architecture Executing Statements Retrieving Results JDBC Drivers JDBC URL's Problems with JDBC	Object Relational Mapping Java Persistence API JPA Versions Entity Classes Entity Manager Persistence Context Entity Identity Entity Lifecycle Entity Relationships Persisting Objects Removing Objects Merging Objects Managing Identity	Mapping Annotations Table Annotation UniqueConstraint Annotation Column Annotation Id Annotation IdClass Annotation GeneratedValue Annotation Version Annotation Basic Annotation Lob Annotation Temporal Annotation Enumerated Annotation Transient Annotation
Module 4 : Mapping Relationships	**Module 5 : Mapping Inheritance**	**Module 6 : JPA Query Language**
Entity Relationship types Bidirectional OneToOne Bidirectional ManyToOne Bidirectional OneToMany Bidirectional ManyToMany Unidirectional OneToOne Unidirectional ManyToOne Unidirectional OneToMany Unidirectional ManyToMany Cascading Persist Cascading Merge Cascading Remove	Mapping Class Hierarchies Mapping Strategies Single Table Inheritance DiscriminatorColumn Settings Single Table per Hierarchy Single Table Mapping Features Joined Subclass Strategy InheritanceType Joined Joined Subclass Per Hierarchy Table per Concrete Class Abstract Entity Classes Mapped Superclasses	Java Persistence QL JPA QL Characteristics Query Interface Projections Subqueries Joins Update and Delete Queries Dynamic and Static Queries Criteria API Query Error Detection CriteriaBuilder Metamodel in JPA
Module 7 : Callbacks and Listeners	**Module 8 : Interceptors**	**Module 9 : Java EE integration**
Life Cycle Callback methods Entity Listeners Life Cycle Callback Rules Signature Life Cycle Callbacks Signature Entity Listeners @PrePersist and @PostPersist @PreRemove and @PostRemove @PreUpdate and @PostLoad Multiple Invocation Callbacks Invocation Order	Interceptor Invocation Model @Interceptor Annotation Interceptor Classes Invocation Context @AroundInvoke Interceptor Lifecycle Interceptor Types Default and Exclude Interceptors PostConstruct or PostActivate PreDestroy and PrePassivate	Enterprise Java Beans Sessions Beans Statefull and Stateless JNDI lookups EJB injection Transaction-Scoped Persistence Context Extended Persistence Context Persistence Unit Packaging in EAR files Deployment Descriptors
Module 10 : Transactions		
Data Integrity Transaction Control Begin, Commit and Rollback Demarcating Boundaries Container Managed Bean Managed Client Managed Transaction Attributes SessionSynchronization JTA Transactions Before Completion After Completion		

SpiralTrain BV
Standerdmolen 10, 2e verdieping
3995 AA Houten

info@spiraltrain.nl
www.spiraltrain.nl
Tel.: +31 (0) 30 – 737 0661

Locations
Houten, Amsterdam, Rotterdam, Eindhoven, Zwolle, Online