

## Programme de formation

### Hibernate

Simplifiez l'accès à vos données persistantes en Java avec Hibernate

#### Durée

3 jours / 21 heures

#### Description

Hibernate est un framework open source permettant de gérer plus simplement la persistance des objets en base de données relationnelle. Après une présentation de la notion de mapping objet-relationnel et des différentes solutions du marché, cette formation pratique présente la gestion de la persistance avec Hibernate et JPA, ainsi que l'utilisation approfondie de ce framework pour développer avec efficacité une couche de persistance.

#### Objectifs pédagogiques

- Comprendre le rôle de l'API standard JPA pour la gestion des données relationnelles dans un programme Java
- Comprendre l'architecture d'Hibernate
- Savoir mettre en oeuvre les correspondances (« mapping ») Java/Hibernate (identités, héritage, associations)
- Savoir effectuer des requêtes sur les objets
- Savoir gérer les transactions (verrou, concurrence, ...)
- Savoir gérer des cas complexes

#### Public

Développeur Java, Responsables techniques, Chefs de projets (web), Architectes

#### Pré-requis

Connaitre JAVA ou avoir suivi la formation Java initiation

#### Méthodes pédagogiques

50 % théorie / 50 % pratique

#### Profil intervenant

L'ensemble de nos formations sont animées par des formateurs expérimentés possédant une expérience terrain éprouvée.

#### Modalités d'évaluation

L'évaluation des acquis se fait tout au long de la session au travers d'ateliers de mise en pratique des notions et concepts abordés pendant la formation.

## Programme

### Introduction à la problématique de persistance

*Ce chapitre a pour objectif de poser le problème de la persistance d'objets Java et de l'illustrer à l'aide d'un premier TP mettant en oeuvre seulement JDBC.*

- Origines du problème
- Les facettes de l'impédance mismatch
- Architectures de la persistance
- Historique des solutions (JDBC, JDO, Hibernate)
- Rappel sur JDBC / SQL et ses limites

### Travaux Pratiques

*Réalisation du TP1 (persistance à l'aide de JDBC suivant le pattern DAO)*

### Premier pas avec Hibernate - mapping de l'identité

*Ce chapitre présente le minimum à savoir pour commencer à coder une application avec Hibernate. Il permet d'illustrer comment Hibernate résout les problèmes mis en avant dans le TP précédent.*

- Installer Hibernate
- Initialiser le framework Hibernate
- Opérations CRUD sur les objets
- Identité d'objet et cycle de vie

### Travaux Pratiques

*Réalisation du TP2 (reprise du TP1 avec Hibernate)*

### Mapping de l'héritage

*Ce chapitre présente comment Hibernate résout le problème de l'Héritage du modèle objet*

- Le problème du mapping de l'héritage
- Notion de relations polymorphes
- Approche : une table par classe concrète
- Approche : une table par hiérarchie
- Approche : une table par classe

### Travaux Pratiques

*Réalisation du TP3 (enrichissement du TP2 avec l'héritage)*

### Mapping des associations

*Ce chapitre fait le tour des différentes cardinalités d'associations ainsi que de la sémantique de la composition.*

- Association et navigation.
- Association 1-N, N-1, N-N
- Composition

- Classe association

## Travaux Pratiques

*Réalisation du TP4 (enrichissement du TP3 avec plusieurs types d'associations)*

### Accès en lecture aux objets (HQL, API Criteria et requêtes SQL)

*Ce chapitre présente l'important pattern du Lazy loading qui permet de gérer la quantité d'objets chargée par les requêtes en base. Il présente ensuite le langage de requête propre à Hibernate : HQL.*

- Le pattern Lazy Loading
- Usage de HQL
- Usage de l'API Criteria
- Usage des SQL Query

## Travaux Pratiques

*Réalisation du TP5 (requêtes bâties sur le TP4)*

### Accès en lecture aux objets (HQL, API Criteria et requêtes SQL)

*Ce chapitre expose comment sauvegarder des grappes d'objets et comment gérer la propagation de cette sauvegarde.*

- Principe de persistance transitive
- Attachement / détachement des objets d'une session
- Supprimer une grappe d'objets

## Gestion des transactions

*L'objectif de ce chapitre est d'effectuer un rappel sur les notions de transactions en base de données et de transactions utilisateurs.*

- Rappel sur la notion de transaction
- Transaction et concurrence d'accès
- Niveaux d'isolation
- Stratégies de contrôle de la concurrence
- Gestion optimiste
- Gestion pessimiste
- Notion de transaction utilisateur

## Architecture et Hibernate

*L'objectif de ce chapitre est d'exposer un certain nombre de bonnes pratiques d'Hibernate. Plusieurs de ces pratiques vont prendre la forme d'une classe utilitaire expliquée dans le cours et mise en oeuvre dans le TP. Architecture d'Hibernate.*

- Comment gérer la session Hibernate (pattern du ThreadLocal)
- Comment programmer des règles métiers (frontière métier / accès aux données)
- Usage du Pattern Data Access Object (DAO) avec Hibernate
- Comment gérer les transactions utilisateurs avec Hibernate
- Intégrer Hibernate dans une application web

**Travaux Pratiques**

*Réalisation du TP6 (Cas d'utilisation complet)*